

Part²GS: Part-aware Modeling of Articulated Objects using 3D Gaussian Splatting

Tianjiao Yu, Vedant Shah, Muntasir Wahed, Ying Shen, Kiet A. Nguyen, Ismini Lourentzou

{ty41, vrshah4, mwahed2, ying22, kietan2, lourent2}@illinois.edu

University of Illinois Urbana-Champaign

Abstract. Articulated objects are common in the real world, yet modeling their structure and motion remains a challenging task for 3D reconstruction methods. In this work, we introduce **Part²GS**, a novel framework for modeling articulated digital twins of multi-part objects with high-fidelity geometry and physically consistent articulation. Part²GS leverages a part-aware 3D Gaussian representation that encodes articulated components with learnable attributes, enabling structured, disentangled transformations that preserve high-fidelity geometry. To ensure physically consistent motion, we propose a motion-aware canonical representation guided by physics-based constraints, including contact enforcement, velocity consistency, and vector-field alignment. Furthermore, we introduce a field of repel points to prevent part collisions and maintain stable articulation paths, significantly improving motion coherence over baselines. Extensive evaluations on both synthetic and real-world datasets show that Part²GS consistently outperforms state-of-the-art methods by up to 10 \times in Chamfer Distance for movable parts.

🌐 <https://plan-lab.github.io/part2gs>

1. Introduction

Articulated objects, structures composed of multiple rigid parts connected via joints, are pervasive in real-world environments and play a central role in physical interaction and manipulation tasks. Creating 3D assets that represent articulated objects is highly valuable for a variety of applications in 3D perception [3, 4, 6, 11, 23, 29], embodied AI [2, 15, 37], and robotics [5, 36, 38]. Despite their clear utility across these research domains, most available articulated 3D assets are created manually, and existing datasets are often limited in both scale and diversity [24, 27], restricting advancements that can effectively understand and manipulate articulated objects in diverse, real-world environments. To address this challenge, recent efforts have focused on reconstructing articulated objects from real-world observations [8, 42] or predicting articulation patterns for existing 3D models [17, 25]. However, these methods often rely on labor-intensive data collection processes or large, predefined datasets of 3D objects with detailed geometry.

Recently, generative models have made progress in articulated 3D object reconstruction by leveraging 3D Gaussian Splatting (3DGS) or Neural Radiance Fields (NeRFs) [7, 30, 42, 43]. While effective, these approaches largely treat articulated motion as a geometric interpolation problem, without incorporating physical feasibility or semantic part understanding. This often results in reconstructions that lack groundedness, such as floating components or physically implausible joint behavior, especially when dealing with complex, multi-part objects. Moreover, existing methods rely on direct state-to-state interpolation and clustering, which fail to account for rigid-body coherence or articulation constraints in unconstrained scenarios [16, 30].

To overcome these limitations, we propose **Part-aware Object Articulation with 3D Gaussian Splatting (Part²GS)**, a novel framework that tackles three core challenges in articulated object modeling: (1) **Unstructured Part Articulation**: Rather than relying solely on unsupervised clustering, dual-quaternion blending, or using predefined part ground truth, Part²GS introduces a part parameter into the standard Gaussian parameters, and guides part transformation with physics-aware forces and learned part embeddings. (2) **No Physical Constraints**: Existing methods lack grounding, collision avoidance, and coherent rigid-body

* Preprint. Work in progress.

motion, resulting in implausible part behavior [25, 26]. Part²GS integrates a physically motivated construction loss that incorporates contact constraints, velocity consistency, and vector-field alignment to ensure stable, realistic articulation. **(3) Rigid State-Pair Modeling:** Prior methods rely heavily on fixed, geometric interpolation between two states [24, 30, 47]. In contrast, Part²GS builds a canonical representation via motion-informed interpolation and is optimized with part-disentangled dynamics, allowing more flexible and physically grounded articulation learning without requiring explicit part supervision.

Through extensive experiments, we demonstrate that Part²GS achieves state-of-the-art performance in reconstructing articulated 3D objects, delivering high-fidelity geometry and physically consistent motion, even in challenging multi-part scenarios. Our contributions are summarized as follows:

- (1) We introduce **Part²GS**, a part-aware 3D Gaussian representation for articulated object reconstruction, that encodes object parts with learnable attributes, enabling disentangled part motions and producing high-fidelity geometry with physically consistent articulation, even in complex multi-part settings.
- (2) We develop a **motion-aware canonical representation** that leverages physics-guided learning to model object articulation with contact constraints, such as velocity consistency and vector-field alignment, while a field of **repel points** pushes parts for better articulation learning. Together, these elements yield part-disentangled geometry and physically plausible motion paths.
- (3) We extensively evaluate Part²GS on both synthetic and real-world articulated objects, achieving state-of-the-art performance over strong baselines. Comprehensive ablations confirm the effectiveness of each component in delivering high-quality geometry and articulation.

2. Related Works

Articulated Object Modeling. Early work on articulated object modeling relied entirely on geometric reasoning and heuristic rules. Given a mesh, slippage analysis and probing techniques were used to detect rotational and translational axes by observing when two parts penetrate or slip past each other [50], and joint types and limits were set by trial-and-error bisection [18, 35, 40]. More recently, supervised learning has taken center stage. Methods that canonize parts into normalized coordinate spaces at both the object and part levels learn to map arbitrary poses to a template frame, then recover joints by fitting rigid transforms [6, 9, 20]. To reduce reliance on labeled data, self-supervised methods learn correspondence or reconstruction consistency instead of explicit annotations. By tracking points across frames and fitting trajectories, one can infer axes and limits without manual labels [41]. More recent single-frame methods warp parts to and from learned canonical spaces to extract joint transforms via reconstruction loss [24, 29]. Despite these advances, such methods still depend on external resources, e.g., predefined part libraries, kinematic graphs, or category-specific templates [12, 17, 25, 26]. In contrast, Part²GS recovers part decompositions and articulation parameters directly from raw multi-view observations, without assuming any prior structural knowledge or category-specific models.

Dynamic Gaussian Modeling. Building on seminal work [14], a wave of follow-up research has extended 3D Gaussian Splatting into 4D reconstruction, e.g., learning per-Gaussian deformation fields for animatable human avatars [13], or smoothly interpolating Gaussian attributes over time to replay dynamic scenes [48]. At the same time, methods have been proposed to reinforce both temporal coherence and geometric detail-preserving Gaussian identities across frames to stabilize synthesis, embedding temporal features for live novel-view rendering, and enforcing geometry-aware deformations that conform to local surface structure [21, 31, 32, 44]. A complementary line of research has extended Gaussian Splatting for fully animatable avatars, learning per-splat pose controls, disentangling distinct motion modes, and even dispensing with

prebuilt templates to allow free reposing of arbitrary scenes [1, 39, 46]. At the same time, sparse “superpoint” formulations have been introduced to give users direct, real-time editing of Gaussian clusters, trading off physics or kinematic structure recovery in favor of interactive deformability [10, 45]. We build on these advances by introducing part-aware dynamic Gaussian modeling, linking motion to discovered part structures to achieve fine-grained, controllable motion synthesis without object-specific priors or templates, ensuring collision-free articulation and consistent part-based transformations.

3. Preliminaries

3D Gaussian Splatting. 3D Gaussian Splatting (3DGS) [14] is a recent state-of-the-art approach for representing 3D scenes by parameterizing them as collections of anisotropic Gaussians. Unlike implicit representation methods such as NeRF [34], which relies on volume rendering, 3DGS achieves real-time rendering by splatting these Gaussians onto a 2D plane and compositing their effects through differentiable alpha blending [51]. Formally, a scene is modeled as a set of N anisotropic Gaussians, denoted as

$$\mathcal{G} = \{G_i : \mu_i, r_i, s_i, \sigma_i, h_i\}_{i=1}^N, \quad (1)$$

where each Gaussian G_i is parameterized by its centroid position $\mu_i \in \mathbb{R}^3$, rotation quaternion $r_i \in \mathbb{R}^4$, anisotropic scale vector $s_i \in \mathbb{R}^3$, scalar opacity $\sigma_i \in [0, 1]$, and spherical harmonics coefficients h_i that encode view-dependent appearance. The opacity value of a Gaussian G_i at any spatial point $x \in \mathbb{R}^3$ is computed as:

$$\alpha_i(x) = \sigma_i \exp\left(-\frac{1}{2}(x - \mu_i)^\top \Sigma_i^{-1}(x - \mu_i)\right). \quad (2)$$

The covariance matrix Σ_i characterizing the anisotropic spread of the Gaussian is defined as $\Sigma_i = R_i S_i S_i^\top R_i^\top$. Here, S_i is a diagonal matrix of scaling factors, and R_i is a rotation matrix corresponding to quaternion r_i . This decomposition ensures that the covariance matrix remains positive semi-definite, maintaining a valid geometric interpretation of Gaussian spread and orientation.

To render a scene represented by the set of Gaussians \mathcal{G} , we need to project them onto a 2D plane. The projection is achieved using differentiable α -blending, which combines their opacity and spherical harmonic-based color contributions. Formally, the rendering equation for image I is defined as:

$$I = \sum_{i=1}^N T_i \alpha_i^{\mathbb{R}^2} \mathcal{H}(h_i, v_i), \quad \text{where} \quad T_i = \prod_{j=1}^{i-1} (1 - \alpha_j^{\mathbb{R}^2}). \quad (3)$$

Here, $\alpha_i^{\mathbb{R}^2}$ is the projected 2D Gaussian opacity evaluated at each pixel coordinate, analogous to its 3D counterpart. The term $\mathcal{H}(h_i, v_i)$ represents the spherical harmonics-based color function evaluated along viewing direction v_i , while the blending weights T_i encode front-to-back occlusion and transparency effects. Given N multi-view images $\mathcal{I} = \{I_i\}_{i=1}^N$, Gaussian parameters \mathcal{G} are optimized by minimizing rendering loss:

$$\mathcal{L}_{\text{render}} = (1 - \lambda) \mathcal{L}_I + \lambda \mathcal{L}_{\text{D-SSIM}}, \quad (4)$$

where $\mathcal{L}_I = ||I - \bar{I}||_1$ is the pixel-wise ℓ_1 reconstruction loss, $\mathcal{L}_{\text{D-SSIM}}$ measures the perceptual structural similarity between rendered and target images [14], and λ is the loss coefficient. This explicit Gaussian-based scene representation, combined with a differentiable rendering process, enables efficient inference of the 3D structure directly from view-based supervision.

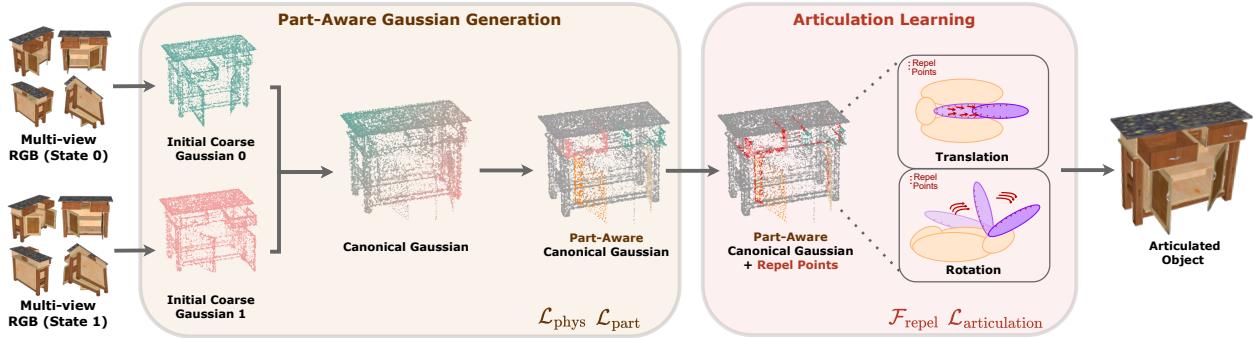


Figure 1: Overview of Part²GS. Given two sets of multi-view images of an object, we first reconstruct independent coarse 3D Gaussian models and learn a motion-informed, part-aware canonical Gaussian. We optimize the canonical Gaussian under physical constraints $\mathcal{L}_{\text{phys}}$ and part awareness $\mathcal{L}_{\text{part}}$. Finally, we learn the articulation model with repel points $\mathcal{F}_{\text{repel}}$ and $\mathcal{L}_{\text{articulation}}$ (details in §4).

4. Part²GS: Part-aware Object Articulation with 3D Gaussian Splatting

We introduce **Part²GS**, a method that constructs articulated 3D object representations by leveraging 3D Gaussian Splatting for part-aware geometry and articulation learning. Given a set of 2D multi-view images $\mathcal{I}_t = \{\mathcal{I}_i^t\}_{i=1}^N$ collected at two distinct joint states $t \in \{0, 1\}$, our objective is to generate an articulated 3D object representation \mathcal{O} with part-level disentanglement and physically grounded motion. \mathcal{O} is modeled as a composition of a static base $\mathcal{G}_{\text{static}}$ and K movable parts, represented as $\mathcal{G} = \{\mathcal{G}_{\text{static}}, \mathcal{G}_k \mid k \in [1, \dots, K]\}$. Each part \mathcal{G}_k is modeled as a collection of M_k 3D Gaussians $\mathcal{G}_k = \{G_i^k \mid i \in [1, \dots, M_k]\}$, enabling flexible manipulation and clear part delineation.

Part²GS consists of two main stages: **(1) Part-Aware Gaussian Generation** (§4.1): We first generate canonical Gaussian representations independently for each joint state ($\mathcal{I}_0, \mathcal{I}_1$) and subsequently infer per-Gaussian part identities implicitly through multi-view geometric cues. Specifically, each Gaussian G_i is augmented with a compact, learnable part-identification parameter ψ_i that enables unsupervised clustering of Gaussians into meaningful, physically consistent parts. **(2) Articulation Learning** (§4.2): Once the part-aware canonical Gaussian representation is learned, we model the transformations of each articulated part using SE(3) rigid body motions. To enforce physically plausible and collision-free motion, we introduce *repel points*, distributed across the affinity between different parts. These repel points apply localized repulsive forces that prevent unrealistic part overlap during articulation and act as initialization guides for smooth trajectory optimization, ensuring smooth and stable motion paths throughout articulation (Figure 1).

4.1. Part-Aware Gaussian Generation

Coarse Gaussian Initialization. We begin by independently optimizing two sets of single-state Gaussians, $\mathcal{G}_{\text{single}}^0$ and $\mathcal{G}_{\text{single}}^1$, using multi-view images captured at two distinct joint states, \mathcal{I}^0 and \mathcal{I}^1 accordingly. Each set is optimized by minimizing the differentiable rendering loss described in Eq. (4). Prior approaches that rely on directly modeling correspondences between two distinct states often suffer from severe occlusion, viewpoint inconsistencies, and difficulties arising from learning articulation deformation while maintaining rigid geometry [12, 47]. To address these challenges, we propose constructing a canonical intermediate representation that bridges the two observed states. We first establish correspondences between $\mathcal{G}_{\text{single}}^0$ and $\mathcal{G}_{\text{single}}^1$ via Hungarian matching based on pairwise distances between Gaussian centers. For each matched

pair, rather than simply averaging [30], we create a canonical Gaussian by interpolating between the two corresponding Gaussians. Specifically, we introduce a *motion-informed prior* to guide the interpolation. We estimate the motion richness of each state by computing the mean minimum distance from each Gaussian in one state to its nearest neighbor in the other state. Formally, for each state $t \in \{0, 1\}$, we compute:

$$D^{t \rightarrow \bar{t}} = \mathbb{E}_i \left[\min_j \| \boldsymbol{\mu}_i^{(t)} - \boldsymbol{\mu}_j^{(1-t)} \|_2 \right], \quad \text{where } \bar{t} = 1 - t \text{ denotes the opposite state.} \quad (5)$$

The state with the higher $D^{t \rightarrow \bar{t}}$ value is identified as the *motion-informative state*, reflecting greater articulation or part displacement. We then bias the interpolation toward the motion-informative state when constructing the canonical Gaussians. For a matched Gaussian pair (G_i^0, G_i^1) , the canonical Gaussian G_i^c is computed as:

$$\boldsymbol{\mu}_i^c = (1 - \beta) \boldsymbol{\mu}_i^0 + \beta \boldsymbol{\mu}_i^1, \quad (6)$$

where $\beta \in [0, 1]$ is a fixed bias coefficient favoring the motion-informative state, with its value determined by the relative displacement magnitudes between the two articulation states. This motion-aware canonical initialization, denoted as $\mathcal{G}_{\text{coarse}}^c$, enables the model to better capture part structures and articulation dynamics in subsequent learning stages.

Part Discovery. To achieve a detailed and controllable representation of articulated objects, it is crucial to explicitly model the object’s semantic decomposition into parts. While standard 3D Gaussian Splatting provides efficient geometric reconstruction, it lacks explicit part-level semantics necessary for articulated object modeling. Motivated by this, we propose to augment each Gaussian representation, introduced in Eq. (1), with a compact, learnable parameter ψ_i that encodes the identity of the part to which it belongs, termed *part-identification parameter*. This part-identification parameter provides a consistent identity for each part across views, making it possible to cluster Gaussians based on their part assignments.

Formally, given K parts, our objective is to compute part-identity embedding ψ_i that assigns each Gaussian \mathcal{G}_i to a specific part for every canonical Gaussian G_i^c . We adopt a cluster-based initialization to group Gaussians according to their part membership, inspired by recent works [10, 30]. Specifically, we define K learnable cluster centers $\mathbf{C}_k = \{\boldsymbol{\mu}_k, \mathbf{R}_k, \mathbf{s}_k\}$ with center location $\boldsymbol{\mu}_k \in \mathbb{R}^3$, rotation matrix $\mathbf{R}_k \in \mathbb{R}^{3 \times 3}$, and scale vector $\mathbf{s}_k \in \mathbb{R}^3$. For a given Gaussian $G_i \in \mathcal{G}^c$, we compute the Mahalanobis distance \mathbf{D}_i^k between G_i and center \mathbf{C}_k :

$$\mathbf{D}_i^k = (\mathbf{X}_i^k)^T \mathbf{X}_i^k, \quad \text{where} \quad \mathbf{X}_i^k = \frac{\mathbf{R}_k(\boldsymbol{\mu}_i^c - \boldsymbol{\mu}_k)}{\mathbf{s}_k}. \quad (7)$$

Here, \mathbf{D}_i^k measures the normalized distance matrix for part assignment. During subsequent optimization, the embeddings ψ_i are treated as free parameters. To maintain spatial and semantic consistency, we introduce a self-supervised consistency loss that penalizes divergence in part embeddings for neighboring Gaussians with similar displacement trajectories:

$$\mathcal{L}_{\text{cons}} = \sum_{i,j} w_{ij} \| \boldsymbol{\psi}_i - \boldsymbol{\psi}_j \|^2, \quad (8)$$

where $w_{ij} = [1 \text{ if } \| \boldsymbol{\mu}_i^c - \boldsymbol{\mu}_j^c \| < r \text{ and } \| \mathbf{d}_i - \mathbf{d}_j \| < \delta] \text{ and } \{\mathbf{d}_i, \mathbf{d}_j\}$ denote the displacement vectors between the two Gaussians. This consistency loss ensures that Gaussians with similar spatial relationships are encouraged to maintain similar part-identity embeddings.

Optimization. We design the overall loss, $\mathcal{L}_{\text{construct}}$, with three components to ensure accurate part grouping and physical plausibility in the 3D representation of the object. The loss components are:

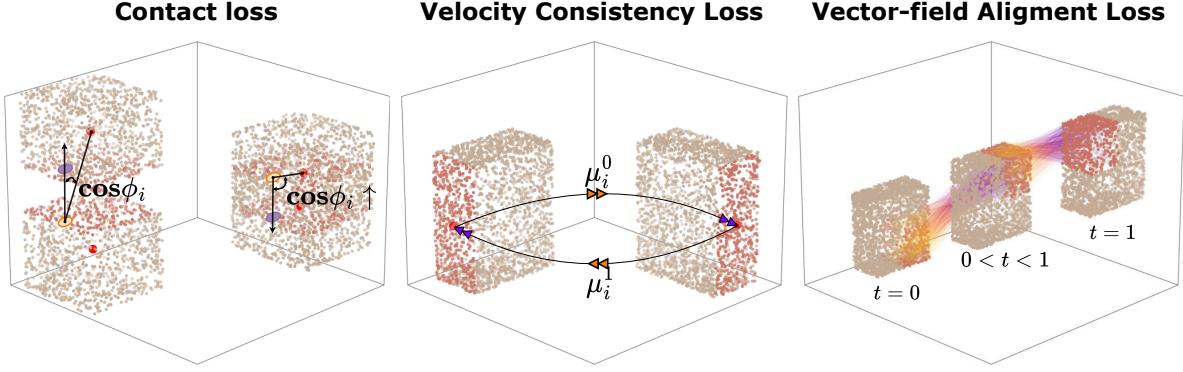


Figure 2: Physical Constraints. (1) *Contact Loss* penalizes interpenetration by minimizing the angle between two vectors for each Gaussian: a) the vector pointing to the center of the opposing part, and b) the vector pointing to its nearest Gaussian in that part. Red dots (\bullet) denote object centers. (2) *Velocity Consistency* encourages coherent motion trajectories (e.g., $\mu_i^0 == \mu_i^1$). Red dots (\bullet) represent the same Gaussian at different states. (3) *Vector-field Alignment* enforces consistency between predicted part transformations and observed motions (see **Physical Constraints** in §4.1) for more details.

Part Loss ($\mathcal{L}_{\text{part}}$). To improve grouping accuracy, we introduce a regularization loss that enhances the learning of the part-identity embedding ψ_i , for each 3D Gaussian. This loss enforces 3D spatial consistency by encouraging similar encodings among neighboring Gaussians, even in heavily occluded regions, and is computed as a batch-wise KL divergence:

$$\mathcal{L}_{\text{part}} = \frac{1}{M} \sum_{i=1}^M D_{\text{KL}} \left(F(G_i) \parallel \frac{1}{|\mathcal{N}(G_i)|} \sum_{j \in \mathcal{N}(G_i)} F(G_j) \right), \quad (9)$$

where M is the number of Gaussians in the current batch, $F(G_i) = \text{softmax}(f(\psi_i))$ is the part identity probability distribution for each Gaussian G_i , computed by projecting part-identity encodings into K part categories through a shared linear layer f followed by a softmax operation.

Physical Constraints ($\mathcal{L}_{\text{phys}}$). To preserve the physical plausibility of articulated motion, we incorporate three auxiliary losses that constrain part-level deformation: contact loss, vector-field alignment, and velocity consistency (Figure 2). First, the **contact loss** discourages unrealistic interpenetration between movable parts and the static base by introducing a contact-based constraint. For each Gaussian center $\mu_i \in G_i^k$ belonging to a movable part \mathcal{G}_k , we identify and locate its nearest corresponding static Gaussian center μ_i^* . Let $\bar{\mu}$ be the centroid of the static base $\mathcal{G}_{\text{static}}$, and define $\mathbf{d}_i = \mu_i - \mu_i^*$, $\mathbf{d}_k = \mu_i - \bar{\mu}$, where \mathbf{d}_i represents the offset from the movable part to its nearest static Gaussian, and \mathbf{d}_k captures the displacement from the movable part to the centroid of the static base. The cosine of the angle φ_i between these two vectors penalizes obtuse contact angles via:

$$\mathcal{L}_{\text{contact}} = \frac{1}{|\mathcal{G}_k|} \sum_{i \in \mathcal{G}_k} \max(0, -\cos \varphi_i), \quad \text{where} \quad \cos \varphi_i = \frac{\mathbf{d}_i^\top \mathbf{d}_k}{\|\mathbf{d}_i\| \|\mathbf{d}_k\|}. \quad (10)$$

Since rigid parts should exhibit coherent motion, we employ a **velocity consistency loss** [19, 22, 28] by

defining per-Gaussian displacements $\Delta\boldsymbol{\mu}_i = \boldsymbol{\mu}_i^1 - \boldsymbol{\mu}_i^0$, and penalizing intra-part variance:

$$\mathcal{L}_{\text{velocity}} = \sum_{k=1}^K \text{Var}(\{\Delta\boldsymbol{\mu}_i \mid i \in \mathcal{G}_k\}). \quad (11)$$

We additionally employ a **vector-field alignment loss** to ensure that predicted part transformations remain consistent with observed motion across different joint states. Inspired by flow-based models [19, 22, 28], we treat part articulation as an SE(3) vector field acting on canonical Gaussians. For each part transformation $T_k = (\mathbf{R}_k, \mathbf{t}_k) \in \text{SE}(3)$, we enforce consistency between predicted and observed positions:

$$\mathcal{L}_{\text{vector}} = \sum_{k=1}^K \sum_{i \in \mathcal{G}_k} \left\| \mathbf{R}_k \boldsymbol{\mu}_i^0 + \mathbf{t}_k - \boldsymbol{\mu}_i^1 \right\|^2. \quad (12)$$

These physical constraints provide a simple yet effective barrier against self-collision while keeping per-part motion rigid and coherent. The combined loss is $\mathcal{L}_{\text{phys}} = \mathcal{L}_{\text{contact}} + \mathcal{L}_{\text{velocity}} + \mathcal{L}_{\text{vector}}$.

Rendering Loss ($\mathcal{L}_{\text{render}}$). The overall Gaussian parameters are optimized by minimizing Eq. (4).

The total construction loss $\mathcal{L}_{\text{construct}}$ is thus formulated as:

$$\mathcal{L}_{\text{construct}} = \mathcal{L}_{\text{render}} + \lambda_{\text{part}} \mathcal{L}_{\text{part}} + \lambda_{\text{phys}} \mathcal{L}_{\text{phys}}, \quad (13)$$

where λ_{part} and λ_{phys} are weights for the part loss and physical loss, respectively. This combined loss promotes accurate part grouping while enforcing a stable and physically realistic 3D configuration, resulting in a robust and controllable 3D representation of the object \mathcal{O} .

4.2. Articulation Learning

To enable realistic articulation of the object’s movable parts relative to its static base, we introduce *repel points* distributed across the static base, $\mathcal{R} = \{\mathbf{r}_j \in \mathbb{R}^3 \mid j = 1, 2, \dots, N_R\}$, where N_R is the total number of repel points, and each \mathbf{r}_j is associated with a repulsion field that encourages each movable part to find a stable configuration while avoiding excessive overlap with the static base. These repel points, placed in regions of articulated parts where the static and movable parts are initially close, apply localized repulsive forces that guide the part’s movement while maintaining physical separation. Repulsion force is defined as

$$\mathbf{F}_{\text{repel},i}^k = \sum_{\mathbf{r}_j \in \mathcal{R}} k_r \cdot \frac{(\mathbf{r}_j - \boldsymbol{\mu}_i^k)}{\|\mathbf{r}_j - \boldsymbol{\mu}_i^k\|^3}, \quad (14)$$

where k_r is a repulsion coefficient, $\boldsymbol{\mu}_i$ is the center of the Gaussian G_i , \mathbf{r}_j is the j -th repeller point, and $\mathbf{F}_{\text{repel},i}^k$ is the force vector applied to Gaussian G_i^k .

Once repel points are established, we optimize transformations that capture feasible movement trajectories of each movable part relative to the static base. We define a transformation for each movable part, represented by $T_k = (\mathbf{R}_k, \mathbf{t}_k) \in \text{SE}(3)$, where R_k is the rotation matrix and $t_k \in \mathbb{R}^3$ denotes the translation vector of the k -th movable part with respect to the static base. To learn the true movement, we initialize with random transformations $T_k^{(0)} = (\mathbf{R}_k^{(0)}, \mathbf{t}_k^{(0)})$ and iteratively refine them by aligning the predicted positions of the Gaussian centers with their observed locations during articulation. Specifically, at each iteration step t , the transformed position of each Gaussian G_i^k under the current transformation is calculated as

$\mu_i^{k,(t)} = \mathbf{R}_k^{(t)} \mu_i^{k,0} + \mathbf{t}_k^{(t)}$, where $\mu_i^{k,0}$ is the initial canonical position of the Gaussian. To enforce collision-free motion, each Gaussian is further adjusted based on the influence of nearby repel points:

$$\mu_i^{k,(t)} \leftarrow \mu_i^{k,(t)} + \mathbf{F}_{\text{repel},i}^k. \quad (15)$$

We optimize the part trajectories by minimizing an articulation loss that enforces both positional alignment and rotational consistency at each iteration step t :

$$\mathcal{L}_{\text{articulation}}^{(t)} = \sum_{k=1}^K \sum_{i \in \mathcal{G}_k} \left(\left\| (\mathbf{R}_k^{(t)} \mu_i^{k,0} + \mathbf{t}_k^{(t)} + \mathbf{F}_{\text{repel},i}^k) - \hat{\mu}_i^k \right\|^2 + \lambda_{\text{rot}} \text{Angle}(\mathbf{R}_k^{(t)}, \hat{\mathbf{R}}_k) \right), \quad (16)$$

where λ_{rot} is a weighting factor enforcing rotational alignment and $\text{Angle}(\cdot)$ measures the rotational deviation. Additionally, we leverage the aforementioned contact loss $\mathcal{L}_{\text{contact}}$ and $\mathcal{L}_{\text{part}}$ to prevent the movable part from overlapping with the static base or other parts, ensuring physical plausibility throughout the articulation process. Through this iterative process, we converge on a set of transformations $\mathcal{T} = \{T_k \mid k \in [1, \dots, K]\}$ that capture realistic movement paths of each movable part with respect to the static base. This articulation learning framework, grounded in repel points, transformation refinement, and contact-aware constraints, provides a robust model for representing and manipulating the articulated parts of the object \mathcal{O} .

5. Experiments

We compare **Part²GS** against Ditto [12], PARIS [24], ArtGS [30], and DTA [47] on three object articulation datasets with varying levels of articulation complexity: PARIS [24] (10 synthetic objects with 1 movable part), ARTGS-MULTI [30] (5 synthetic objects with 3–6 movable parts), and DTA-MULTI [47] (2 synthetic objects with 2 movable parts). Following prior articulated object modeling work [12, 24, 30], to assess geometric quality, we report Chamfer Distance scores separately for the entire object (CD_{whole}), the static components (CD_{static}), and the average of the movable parts (CD_{movable}). To assess articulation accuracy, we measure the angular deviation between the predicted and actual joint axes (Ang Err), the positional offset for revolute joints (Pos Err), and the part motion error (Motion Err).

5.1. Experimental Results

Table 1 presents results on the PARIS benchmark, where our method, Part²GS, consistently achieves the lowest angular and positional errors, demonstrating superior joint parameter estimation. In particular, the average angular error remains at or below 0.01° across all simulated objects, significantly outperforming Ditto [12] and PARIS [24], whose errors range from several degrees to near-degenerate predictions. Positional accuracy for revolute joints similarly favors our method, achieving near-zero errors on all relevant instances. On motion execution accuracy, measured by geodesic or Euclidean distance depending on joint type, Part²GS also leads with near-zero error on most categories, closely followed by ArtGS [30]. This highlights the benefit of our motion-consistent design. In addition, in terms of geometry, Part²GS consistently maintains better geometric fidelity, reducing Chamfer Distance across all categories by up to $1.74\times$ compared to the next best baseline, and showing a $2\text{--}4\times$ improvement over DTA and ArtGS on both static and dynamic geometry. Notably, our performance on the most challenging metric CD_{movable} demonstrates that Part²GS not only captures the geometry of moving parts accurately but also preserves their structural integrity during articulation, something that competing methods struggle to achieve.

Table 1: Quantitative Results on PARIS. Lower (\downarrow) is better across all metrics. highlights best performing results. Objects with * are seen categories trained in Ditto. F indicates wrong motion type predictions. Pos Err is omitted for objects with only prismatic joints (Blade, Storage*, and Real-Storage).

Metric	Method	Simulation										Real	
		Foldchair	Fridge	Laptop*	Oven*	Scissor	Stapler	USB	Washer	Blade	Storage*	Real-Fridge	Real-Storage
Motion	Ditto	89.35	89.30	3.12	0.96	4.50	89.86	89.77	89.51	79.54	6.32	1.71	5.88
	PARIS	19.05	7.87	0.03	9.21	22.34	8.89	0.82	22.18	50.45	0.03	9.92	77.83
	DTA	0.03	0.09	0.07	0.22	0.10	0.07	0.11	0.36	0.20	0.09	2.08	13.64
	ArtGS	0.01	0.03	0.01	0.01	0.05	0.01	0.04	0.02	0.03	0.01	2.09	3.47
	Part ² GS (Ours)	0.01	0.01	0.01	0.01	0.02	0.01	0.01	0.01	0.01	0.02	0.03	1.24
Pos Err	Ditto	3.77	1.02	0.01	0.13	5.70	0.20	5.41	0.66	-	-	1.84	-
	PARIS	0.35	3.13	0.04	0.07	2.59	7.67	6.35	4.05	-	-	1.50	-
	DTA	0.01	0.01	0.01	0.01	0.02	0.02	0.00	0.05	-	-	0.59	-
	ArtGS	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	-	-	0.47	-
	Part ² GS (Ours)	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	-	-	0.13	-
Motion Err	Ditto	99.36	F	5.18	2.09	19.28	56.61	80.60	55.72	F	0.09	8.43	0.38
	PARIS	166.24	102.34	0.03	28.18	124.38	117.71	167.98	126.77	0.38	0.36	2.68	0.58
	DTA	0.10	0.12	0.11	0.12	0.37	0.08	0.15	0.28	0.00	0.00	1.85	0.14
	ArtGS	0.03	0.04	0.02	0.02	0.04	0.01	0.03	0.03	0.00	0.00	1.94	0.04
	Part ² GS (Ours)	0.01	0.01	0.01	0.00	0.01	0.00	0.01	0.02	0.00	0.00	0.72	0.02
Geometry	Ditto	33.79	3.05	0.25	2.52	39.07	41.64	2.64	10.32	46.90	9.18	47.01	16.09
	PARIS	11.21	11.78	0.17	3.58	17.88	4.79	2.41	15.92	2.24	9.83	13.79	23.92
	DTA	0.18	0.62	0.30	4.60	3.55	2.91	2.32	4.56	0.55	4.90	2.36	10.98
	ArtGS	0.26	0.52	0.63	3.88	0.61	3.83	2.25	6.43	0.54	7.31	1.64	2.93
	Part ² GS (Ours)	0.14	0.41	0.15	2.91	0.48	2.36	1.84	3.92	0.42	3.58	1.29	2.12
CD _{static}	Ditto	141.11	0.99	0.19	0.94	20.68	31.21	15.88	12.89	195.93	2.20	50.60	20.35
	PARIS	24.23	12.88	0.17	7.49	18.89	38.42	13.81	379.40	200.24	63.97	91.72	528.83
	DTA	0.15	0.27	0.13	0.44	10.11	1.13	1.47	0.45	2.05	0.36	1.12	30.78
	ArtGS	0.54	0.21	0.13	0.89	0.64	0.52	1.22	0.45	1.12	1.02	0.66	6.28
	Part ² GS (Ours)	0.12	0.18	0.11	0.38	0.51	0.41	1.05	0.39	1.42	0.78	0.55	5.01
CD _{movable}	Ditto	6.80	2.16	0.31	2.51	1.70	2.38	2.09	7.29	42.04	3.91	6.50	14.08
	PARIS	8.22	9.31	0.28	5.44	6.13	9.62	2.14	14.35	0.76	9.62	11.52	38.94
	DTA	0.27	0.70	0.32	4.24	0.41	1.92	1.17	4.48	0.36	3.99	2.08	8.98
	ArtGS	0.43	0.58	0.50	3.58	0.67	2.63	1.28	5.99	0.61	5.21	1.29	3.23
	Part ² GS (Ours)	0.19	0.43	0.20	1.85	0.42	1.45	0.92	3.45	0.35	2.87	1.03	2.78
CD _{whole}	Ditto	7.32	7.32	7.32	7.32	7.32	7.32	7.32	7.32	7.32	7.32	7.32	7.32
	PARIS	7.32	7.32	7.32	7.32	7.32	7.32	7.32	7.32	7.32	7.32	7.32	7.32
	DTA	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
	ArtGS	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
	Part ² GS (Ours)	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01

Table 2 presents results on the DTA-MULTI and ARTGS-MULTI benchmarks, which consist of objects with multiple movable parts. Part²GS consistently outperforms DTA and ArtGS across all objects and metrics. In terms of articulation accuracy, our method achieves the lowest angular and positional errors on nearly all examples. Improvements are especially pronounced in the part motion error, where we match or outperform the strongest baseline (ArtGS) on all objects, including challenging ones such as Table (5 parts) and Storage (7 parts). Results highlight that Part²GS’s motion estimation remains stable even as the number of joints increases, reflecting robust joint decomposition and alignment. In terms of geometry, Part²GS achieves the lowest Chamfer Distance across all static, movable, and whole-object evaluations in nearly every category. The gain is most visible in CD_{movable}, where our part-aware representation significantly reduces error by up to 10× over DTA and 3× over ArtGS. This indicates that our learned parts remain geometrically coherent even under complex deformations, unlike other baselines, which often suffer from part drift or under-segmentation.

5.2. Ablation Results

We conduct ablations to evaluate the impact of four key Part²GS components: part ID parameters, repulsion points, physical constraints, and canonical initialization. We select the two most complex objects, Table (5 parts) and Storage (7 parts), to demonstrate the influence of each component under challenging articulation scenarios. As shown in Table 3, each contributes differently to articulation and geometry quality. Removing the **part parameters** leads to the most severe degradation across both objects. Angular and motion errors spike dramatically (e.g., Ang Err from 0.03 to 0.21 and Motion Err from 0.01 to 7.32 on the Table object),

Table 2: Results on DTA-MULTI and ArtGS-MULTI. Lower (\downarrow) is better across all metrics. highlights best performing results. Pos Err is omitted for objects with only prismatic joints (Table 4 parts).

Category	Metric	Method	Fridge (3 parts)	Table (4 parts)	Table (5 parts)	Storage (3 parts)	Storage (4 parts)	Storage (7 parts)	Oven (4 parts)
Motion	Ang Err	DTA	0.16	24.35	20.62	0.29	51.18	19.07	17.83
		ArtGS	0.01	1.16	0.04	0.02	0.02	0.14	0.04
		Part ² GS (Ours)	0.01	0.08	0.03	0.01	0.01	0.11	0.03
	Pos Err	DTA	0.01	-	4.2	0.04	2.44	0.31	6.51
		ArtGS	0.00	-	0.00	0.01	0.00	0.02	0.01
		Part ² GS (Ours)	0.00	-	0.00	0.00	0.00	0.01	0.01
	Motion Err	DTA	0.16	0.12	30.8	0.07	43.77	10.67	31.80
		ArtGS	0.03	0.00	0.01	0.01	0.03	0.62	0.23
		Part ² GS (Ours)	0.02	0.00	0.01	0.01	0.02	0.55	0.18
Geometry	CD _{static}	DTA	0.63	0.59	1.39	0.86	5.74	0.82	1.17
		ArtGS	0.62	0.74	1.22	0.78	0.75	0.67	1.08
		Part ² GS (Ours)	0.59	0.56	1.18	0.73	0.68	0.61	1.01
	CD _{movable}	DTA	0.48	104.38	230.38	0.23	246.63	476.91	359.16
		ArtGS	0.13	3.53	3.09	0.23	0.13	3.70	0.25
		Part ² GS (Ours)	0.08	1.95	1.85	0.09	0.07	1.83	0.11
	CD _{whole}	DTA	0.88	0.55	1.00	0.97	0.88	0.71	1.01
		ArtGS	0.75	0.74	1.16	0.93	0.88	0.70	1.03
		Part ² GS (Ours)	0.73	0.51	1.10	0.87	0.80	0.63	0.95

while CD_{movable} skyrockets by over 70 \times . This confirms that semantic part disentanglement is foundational to both accurate motion estimation and coherent geometry recovery. Without explicit part identity supervision, the model fails to isolate and track distinct motions, leading to collapsed or entangled reconstructions.

Disabling the **repel points** has a noticeable effect on motion accuracy but limited influence on geometry quality. On the Table object, motion error increases nearly 50 \times (from 0.01 to 0.48), while angular and positional errors also rise, suggesting that the lack of inter-part repulsion leads to ambiguity in part-specific transformations. However, CD_{whole} remains relatively stable, confirming that the Gaussian reconstruction itself is unaffected. The **physical constraints** contribute moderate improvements, particularly in reducing CD_{movable} and motion error. On both objects, removing these constraints leads to visible but not catastrophic performance drops (e.g., Motion Err from 0.55 to 0.04 and CD_{movable} from 1.83 to 4.54 on Storage), indicating that these constraints provide useful geometric regularization but are not the sole factor in driving accuracy. Finally, removing **canonical initialization** results in the most unstable training behavior. Angular error explodes from 0.11 to 22.15 on Storage, and motion error increases by over 35 \times on both objects. These results highlight the importance of starting from a stable, geometry-aligned canonical state to enable robust part tracking and learning. Without it, the model struggles to learn consistent part transformations across views. In summary, our part-aware design is most crucial for capturing semantic structure, while repulsion and physical priors further enhance geometric precision.

5.3. Qualitative Analysis

Figure 3 qualitatively compares part discovery against the best-performing baseline ArtGS. Part²GS produces clean, consistent segmentation across all configurations. In both the start and end states, Part²GS accurately isolates moving parts (e.g., drawers and doors) with minimal leakage. In the canonical state, where motion cues are weakest, ArtGS fails to maintain distinct part groupings, leading to blurred or collapsed

Table 3: Part²GS Ablations on the two most complex objects in our evaluation, Table (5 parts) and Storage (7 parts). Lower (\downarrow) is better on all metrics. ■ shows results with **all Part²GS modules** while ■ highlights severe failures by removing components of our method. Severe failures are defined as metrics that are more than 5 times worse than the full Part²GS for the same object.

Objects	Methods	AngErr	PosErr	MotionErr	CD _{static}	CD _{movable}	CD _{whole}
Table (5 parts)	✗ part parameters	0.21	■ 0.08	7.32	7.35	■ 145.17	3.10
	✗ repel points	0.09	■ 0.16	0.48	1.19	4.82	1.85
	✗ physical constraints	0.05	■ 0.03	0.18	1.32	4.47	1.65
	✗ canonical init	0.14	■ 0.06	6.32	2.47	■ 117.25	2.62
	Part ² GS (all)	■ 0.03	■ 0.00	■ 0.01	■ 1.18	■ 1.85	■ 1.10
Storage (7 parts)	✗ part parameters	0.26	■ 0.11	10.43	2.95	■ 198.67	3.54
	✗ repel points	0.16	■ 0.14	1.32	0.93	■ 7.43	2.04
	✗ physical constraints	0.04	■ 0.05	0.04	1.22	4.54	1.12
	✗ canonical init	22.15	■ 0.93	19.67	0.79	■ 442.32	1.89
	Part ² GS (all)	■ 0.11	■ 0.01	■ 0.55	■ 0.61	■ 1.83	■ 0.63

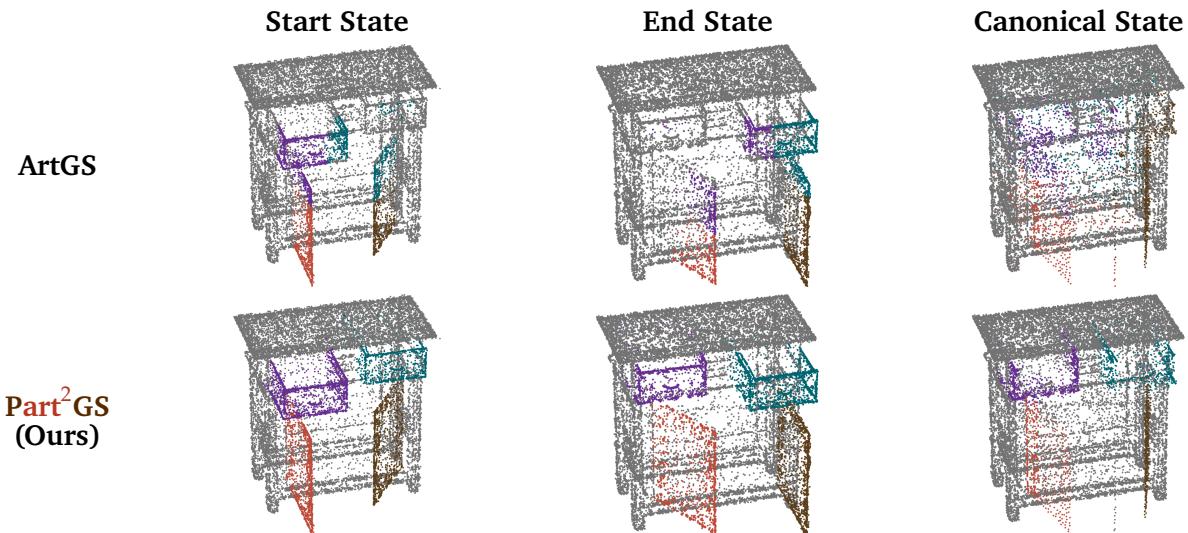


Figure 3: Qualitative Comparison of Part Discovery Across Object States (columns) and Discovery Methods (rows). Part²GS accurately isolates moving parts across start, end, and canonical states.

representations. In contrast, our method Part²GS retains sharp part boundaries, demonstrating robust part identification under challenging intermediate configurations. This highlights the effectiveness of our part-aware training.

6. Conclusions

We introduce Part²GS, a part-aware generative framework for reconstructing articulated digital twins of multi-part objects. By augmenting 3D Gaussians with learnable part attributes and constructing a motion-informed canonical representation, Part²GS enables part decomposition and robust articulation modeling. Through the integration of physical constraint losses and repel points, we enforce grounded, collision-free, and coherent part motion trajectories. Our approach addresses key limitations of prior methods, including lack of semantic part segmentation, absence of physical plausibility, and rigid pose-to-pose modeling assumptions. Experimental results show Part²GS consistently outperforms existing baselines in terms of geometric fidelity and articulation accuracy across diverse categories. Future work could explore extending Part²GS to non-rigid

object articulation and affordance-guided articulation learning.

References

- [1] Jeongmin Bae, Seoha Kim, Youngsik Yun, Hahyun Lee, Gun Bang, and Youngjung Uh. Per-gaussian embedding-based deformation for deformable 3d gaussian splatting. In *European Conference on Computer Vision (ECCV)*, 2024.
- [2] Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Kiana Ehsani, Jordi Salvador, Winson Han, Eric Kolve, Aniruddha Kembhavi, and Roozbeh Mottaghi. Procthor: Large-scale embodied ai using procedural generation. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.
- [3] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [4] Congyue Deng, Jiahui Lei, William B Shen, Kostas Daniilidis, and Leonidas J Guibas. Banana: Banach fixed-point network for pointcloud segmentation with inter-part equivariance. *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.
- [5] Samir Yitzhak Gadre, Kiana Ehsani, and Shuran Song. Act the part: Learning interaction strategies for articulated object part discovery. In *International Conference on Computer Vision (ICCV)*, 2021.
- [6] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [7] Junfu Guo, Yu Xin, Gaoyi Liu, Kai Xu, Ligang Liu, and Ruizhen Hu. Articulatedgs: Self-supervised digital twin modeling of articulated objects using 3d gaussian splatting. *arXiv preprint arXiv:2503.08135*, 2025.
- [8] Nick Heppert, Muhammad Zubair Irshad, Sergey Zakharov, Katherine Liu, Rares Andrei Ambrus, Jeannette Bohg, Abhinav Valada, and Thomas Kollar. Carto: Category and joint agnostic reconstruction of articulated objects. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [9] Ruizhen Hu, Wenchao Li, Oliver Van Kaick, Ariel Shamir, Hao Zhang, and Hui Huang. Learning to predict part mobility from a single static snapshot. *ACM Transactions on Graphics (TOG)*, 2017.
- [10] Yi-Hua Huang, Yang-Tian Sun, Ziyi Yang, Xiaoyang Lyu, Yan-Pei Cao, and Xiaojuan Qi. Sc-gs: Sparse-controlled gaussian splatting for editable dynamic scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
- [11] Ajinkya Jain, Rudolf Lioutikov, Caleb Chuck, and Scott Niekum. Screwnet: Category-independent articulation model estimation from depth images using screw theory. In *International Conference on Robotics and Automation (ICRA)*, 2021.
- [12] Zhenyu Jiang, Cheng-Chun Hsu, and Yuke Zhu. Ditto: Building digital twins of articulated objects from interaction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

-
- [13] HyunJun Jung, Nikolas Brasch, Jifei Song, Eduardo Pérez-Pellitero, Yiren Zhou, Zhihao Li, Nassir Navab, and Benjamin Busam. Deformable 3d gaussian splatting for animatable human avatars. *Computing Research Repository*, 2023.
 - [14] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics (TOG)*, 2023.
 - [15] Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Matt Deitke, Kiana Ehsani, Daniel Gordon, Yuke Zhu, et al. Ai2-thor: An interactive 3d environment for visual ai. *arXiv preprint arXiv:1712.05474*, 2017.
 - [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. In *International Conference on Learning Representations (ICLR)*, 2025.
 - [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 (NeurIPS)*, 2023.
 - [18] Hao Li, Guowei Wan, Honghua Li, Andrei Sharf, Kai Xu, and Baoquan Chen. Mobility fitting using 4d ransac. In *Computer Graphics Forum*, 2016.
 - [19] Sihang Li, Zeyu Jiang, Grace Chen, Chenyang Xu, Siqi Tan, Xue Wang, Irving Fang, Kristof Zyskowski, Shannon P McPherron, Radu Iovita, et al. Garf: Learning generalizable 3d reassembly for real-world fractures. *arXiv preprint arXiv:2504.05400*, 2025.
 - [20] Xiaolong Li, He Wang, Li Yi, Leonidas J Guibas, A Lynn Abbott, and Shuran Song. Category-level articulated object pose estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
 - [21] Zhan Li, Zhang Chen, Zhong Li, and Yi Xu. Spacetime gaussian feature splatting for real-time dynamic view synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
 - [22] Yaron Lipman, Marton Havasi, Peter Holderrieth, Neta Shaul, Matt Le, Brian Karrer, Ricky TQ Chen, David Lopez-Paz, Heli Ben-Hamu, and Itai Gat. Flow matching guide and code. *arXiv preprint arXiv:2412.06264*, 2024.
 - [23] Gengxin Liu, Qian Sun, Haibin Huang, Chongyang Ma, Yulan Guo, Li Yi, Hui Huang, and Ruizhen Hu. Semi-weakly supervised object kinematic motion prediction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
 - [24] Jiayi Liu, Ali Mahdavi-Amiri, and Manolis Savva. Paris: Part-level reconstruction and motion analysis for articulated objects. In *International Conference on Computer Vision (ICCV)*, 2023.
 - [25] Jiayi Liu, Hou In Ivan Tam, Ali Mahdavi-Amiri, and Manolis Savva. Cage: controllable articulation generation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
 - [26] Jiayi Liu, Denys Iliash, Angel X Chang, Manolis Savva, and Ali Mahdavi Amiri. SINGAPO: Single image controlled generation of articulated parts in objects. In *International Conference on Learning Representations (ICLR)*, 2025.

-
- [27] Liu Liu, Wenqiang Xu, Haoyuan Fu, Sucheng Qian, Qiaojun Yu, Yang Han, and Cewu Lu. AKB-48: A real-world articulated object knowledge base. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
 - [28] Xingchao Liu, Chengyue Gong, and qiang liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations (ICLR)*, 2023.
 - [29] Xueyi Liu, Ji Zhang, Ruizhen Hu, Haibin Huang, He Wang, and Li Yi. Self-supervised category-level articulated object pose estimation with part-level SE(3) equivariance. In *International Conference on Learning Representations (ICLR)*, 2023.
 - [30] Yu Liu, Baoxiong Jia, Ruijie Lu, Junfeng Ni, Song-Chun Zhu, and Siyuan Huang. Building interactable replicas of complex articulated objects via gaussian splatting. In *International Conference on Learning Representations (ICLR)*, 2025.
 - [31] Zhicheng Lu, Xiang Guo, Le Hui, Tianrui Chen, Min Yang, Xiao Tang, Feng Zhu, and Yuchao Dai. 3d geometry-aware deformable gaussian splatting for dynamic view synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
 - [32] Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In *International Conference on 3D Vision*, 2024.
 - [33] Yongsen Mao, Yiming Zhang, Hanxiao Jiang, Angel Chang, and Manolis Savva. Multiscan: Scalable rgbd scanning for 3d environments with articulated objects. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.
 - [34] 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*, 2021.
 - [35] Niloy J Mitra, Yong-Liang Yang, Dong-Ming Yan, Wilmot Li, Maneesh Agrawala, et al. Illustrating how mechanical assemblies work. *ACM Transactions on Graphics (TOG)*, 2010.
 - [36] Kaichun Mo, Leonidas J Guibas, Mustafa Mukadam, Abhinav Gupta, and Shubham Tulsiani. Where2act: From pixels to actions for articulated 3D objects. In *International Conference on Computer Vision (ICCV)*, 2021.
 - [37] Xavier Puig, Eric Undersander, Andrew Szot, Mikael Dallaire Cote, Tsung-Yen Yang, Ruslan Partsey, Ruta Desai, Alexander Clegg, Michal Hlavac, So Yeon Min, Vladimír Vondruš, Theophile Gervet, Vincent-Pierre Berges, John M Turner, Oleksandr MakSYMets, Zsolt Kira, Mrinal Kalakrishnan, Jitendra Malik, Devendra Singh Chaplot, Unnat Jain, Dhruv Batra, Akshara Rai, and Roozbeh Mottaghi. Habitat 3.0: A co-habitat for humans, avatars, and robots. In *International Conference on Learning Representations (ICLR)*, 2024.
 - [38] Shengyi Qian and David F Fouhey. Understanding 3d object interaction from a single image. In *International Conference on Computer Vision (ICCV)*, 2023.
 - [39] Zhiyin Qian, Shaofei Wang, Marko Mihajlovic, Andreas Geiger, and Siyu Tang. 3dgs-avatar: Animatable avatars via deformable 3d gaussian splatting. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

-
- [40] Andrei Sharf, Hui Huang, Cheng Liang, Jiapei Zhang, Baoquan Chen, and Minglun Gong. Mobility-trees for indoor scenes manipulation. In *Computer Graphics Forum*, 2014.
 - [41] Yahao Shi, Xinyu Cao, and Bin Zhou. Self-supervised learning of part mobility from point cloud sequence. In *Computer Graphics Forum*, 2021.
 - [42] Chaoyue Song, Jiacheng Wei, Chuan Sheng Foo, Guosheng Lin, and Fayao Liu. Reacto: Reconstructing articulated objects from a single video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
 - [43] Archana Swaminathan, Anubhav Gupta, Kamal Gupta, Shishira R Maiya, Vatsal Agarwal, and Abhinav Shrivastava. Leia: Latent view-invariant embeddings for implicit 3d articulation. In *European Conference on Computer Vision (ECCV)*, 2024.
 - [44] Alexander Vilesov, Pradyumna Chari, and Achuta Kadambi. Cg3d: Compositional generation for text-to-3d via gaussian splatting. *arXiv preprint arXiv:2311.17907*, 2023.
 - [45] Diwen Wan, Ruijie Lu, and Gang Zeng. Superpoint gaussian splatting for real-time high-fidelity dynamic scene reconstruction. In *International Conference on Machine Learning (ICML)*, 2024.
 - [46] Diwen Wan, Yuxiang Wang, Ruijie Lu, and Gang Zeng. Template-free articulated gaussian splatting for real-time reposable dynamic view synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.
 - [47] Yijia Weng, Bowen Wen, Jonathan Tremblay, Valts Blukis, Dieter Fox, Leonidas Guibas, and Stan Birchfield. Neural implicit representation for building digital twins of unknown articulated objects. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
 - [48] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
 - [49] Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao Jiang, Yifu Yuan, He Wang, Li Yi, Angel X. Chang, Leonidas J. Guibas, and Hao Su. SAPIEN: A simulated part-based interactive environment. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
 - [50] Weiwei Xu, Jun Wang, KangKang Yin, Kun Zhou, Michiel Van De Panne, Falai Chen, and Baining Guo. Joint-aware manipulation of deformable models. *ACM Transactions on Graphics (TOG)*, 2009.
 - [51] Wang Yifan, Felice Serena, Shihao Wu, Cengiz Öztireli, and Olga Sorkine-Hornung. Differentiable surface splatting for point-based geometry processing. *ACM Transactions on Graphics (TOG)*, 2019.

A. Experimental Details

Datasets. We conduct evaluations across three distinct datasets, each designed to capture varying levels of articulation complexity: (i) **PARIS** [24], which focuses on objects with simple articulation patterns, specifically those composed of a static base and a single movable part. PARIS includes 10 synthetic examples drawn from the PARTNET-MOBILITY dataset [49] alongside 2 real-world scans collected via the MultiScan [33] system. (ii) **DTA-MULTI** [47], a dataset that offers a moderate challenge with 2 synthetic objects from PARTNET-MOBILITY, where each object contains a static component and two independently movable parts. (iii) **ArtGS-MULTI** [30], a recent dataset that targets more intricate structures, featuring 5 synthetic articulated objects from PARTNET-MOBILITY, each composed of 3 to 6 movable parts, providing a rich testbed for evaluating performance under complex articulation scenarios.

Metrics. To comprehensively evaluate performance, we measure both mesh reconstruction geometry quality and articulation motion accuracy, following common metrics in articulated object modeling [12, 24, 30]. For geometry quality, we calculate the bi-directional Chamfer Distance between the predicted and ground-truth meshes using 10K uniformly sampled points from each. We report Chamfer Distance scores separately for the entire object (CD_{whole}), the static components (CD_{static}), and the average of the movable parts (CD_{movable}). To assess articulation accuracy, we measure the angular deviation between the predicted and actual joint axes (Ang Err), as well as the positional offset for revolute joints (Pos Err). In addition, we report the part motion error (Motion Err), defined as the rotation geodesic distance for revolute joints and Euclidean distance for prismatic ones. Metrics are reported as mean over 10 trials, following prior work [30, 47].

Experiment Setup. For the PARIS dataset, we compare against existing state-of-the-art methods including Ditto [12], PARIS [24], DTA [47], and ArtGS [30]. On the multi-part benchmark, we use DTA and ArtGS as baselines on DTA-MULTI. Following DTA’s protocol [47], we report all metrics as the mean over 3 trials. For multi-part data, where each object may contain many articulated components, we report the average metric across all movable parts. All experiments are conducted on a single RTX 4090 graphics card.

Inference Time. Table 4 compares the per-object inference runtimes of DTA, ArtGS, and our method Part²GS on both simple (one movable part) and complex (multiple movable parts) objects. On the ten simple objects, DTA requires between 28 and 31 minutes each, whereas both ArtGS and Part²GS complete inference in under 10 minutes, yielding roughly a 70–75% speedup. Notably, Part²GS achieves the best or tied-best time on eight out of ten simple objects with ArtGS holding a 1 min edge only on Fridge and Stapler. Despite incorporating additional part-awareness and physical constraints, our method still matches ArtGS’s 8 minute inference performance on most complex objects (and only modestly increases to 10 minutes on the highest-complexity case, Storage₇). Overall, Part²GS delivers state-of-the-art efficiency even with its extra inferential overhead.

Table 4: Inference time for simple and complex objects. Simple objects have one movable part while complex objects have multiple movable parts, denoted by their subscript (e.g., Table₄ has a static base and three movable parts).

Metric	Method	Simple Objects										Complex Objects						
		Foldchair	Fridge	Laptop	Oven	Scissor	Stapler	USB	Washer	Blade	Storage	Fridge ₃	Table ₄	Table ₅	Storage ₃	Storage ₄	Storage ₇	Oven ₄
Time (Min)	DTA	29	30	31	29	28	29	31	28	27	28	32	34	37	32	35	45	35
	ArtGS	9	8	7	7	7	7	7	8	7	8	8	8	8	8	8	8	8
	Part ² GS	8	9	7	8	7	8	7	8	7	9	9	8	9	8	9	10	9

B. Detailed Method Description

B.1. Physical Constraints

Contact Loss. Even in the absence of mesh-based collision resolution, articulated parts should exhibit strict non-penetration constraints. Our contact loss introduces a soft geometric prior that biases part boundaries to remain outside of the static base region. Rather than relying on binary collision detection, we use directional alignment to softly enforce non-intersection via angular consistency [44]. Given a movable part \mathcal{G}_k , for each Gaussian center $\mu_i \in \mathcal{G}_k$, we identify its closest static base Gaussian μ_i^* . Let $\bar{\mu}$ denote the centroid of the static part $\mathcal{G}_{\text{static}}$. Define the vectors: $\mathbf{d}_i = \mu_i - \mu_i^*$, $\mathbf{d}_k = \mu_i - \bar{\mu}$. We compute the cosine angle between these two vectors, and penalize obtuse angles (*i.e.*, where $\cos \varphi_i < 0$) to avoid directional overlap that may indicate intersection:

$$\mathcal{L}_{\text{contact}} = \frac{1}{|\mathcal{G}_k|} \sum_{i \in \mathcal{G}_k} i \max(0, -\cos \varphi_i), \quad \cos \varphi_i = \frac{\mathbf{d}_i^\top \mathbf{d}_k}{\|\mathbf{d}_i\| \|\mathbf{d}_k\|}.$$

As illustrated in Figure 3 (Contact Loss), when movable parts are not intersecting the static base, the contact angle satisfies $\varphi_i < \pi/2$. As the parts move closer and begin to intersect, the angle increases, resulting in $\cos \varphi_i > \pi/2$. By penalizing negative cosine values, the loss encourages contact angles to remain acute, effectively imposing a soft directional repulsion that prevents unrealistic interpenetration between adjacent structures.

Velocity Consistency Loss. Rigid motion implies that all points on a part undergo the same global transformation. While global SE(3) estimation is nontrivial early in training, the local velocity field of each part should still exhibit minimal variance. Enforcing intra-part displacement coherence provides a strong inductive bias for inferring rigid-like motion without hard assignment or part-specific supervision. We observe that rigid transformations induce uniform displacements within each part. Let the displacement vector be $\Delta \mu_i = \mu_i^1 - \mu_i^0$, and penalizing intra-part variance:

$$\mathcal{L}_{\text{velocity}} = \sum_{k=1}^K \text{Var}(\{\Delta \mu_i \mid i \in \mathcal{G}_k\}). \quad (17)$$

This loss promotes uniform translation directions across parts, stabilizing motion inference in early training.

Vector Field Alignment Loss. Learning part-wise transformations in SE(3) is a key challenge in weakly supervised or geometry-driven models. Inspired by dense flow supervision in generative vector field models [19, 22, 28], we treat part articulation as an SE(3) vector field acting on canonical Gaussians. For each part transformation $T_k = (\mathbf{R}_k, \mathbf{t}_k) \in \text{SE}(3)$, we enforce consistency between predicted and observed positions:

$$\mathcal{L}_{\text{vector}} = \sum_{k=1}^K \sum_{i \in \mathcal{G}_k} \left\| \mathbf{R}_k \mu_i^0 + \mathbf{t}_k - \mu_i^1 \right\|^2. \quad (18)$$

These physical constraints provide a simple yet effective barrier against self-collision while keeping per-part motion rigid and coherent. The combined loss is $\mathcal{L}_{\text{phys}} = \mathcal{L}_{\text{contact}} + \mathcal{L}_{\text{velocity}} + \mathcal{L}_{\text{vector}}$. This defines a pointwise alignment loss in Euclidean space \mathbb{R}^3 , treating articulation as a vector field over part-assigned Gaussians.

C. Qualitative Results

Figure 4 illustrates intermediate articulation results across five time steps ($T = \{0, 0.25, 0.5, 0.75, 1\}$) for three distinct articulated objects with varied joint types and geometries. Each row shows a different object

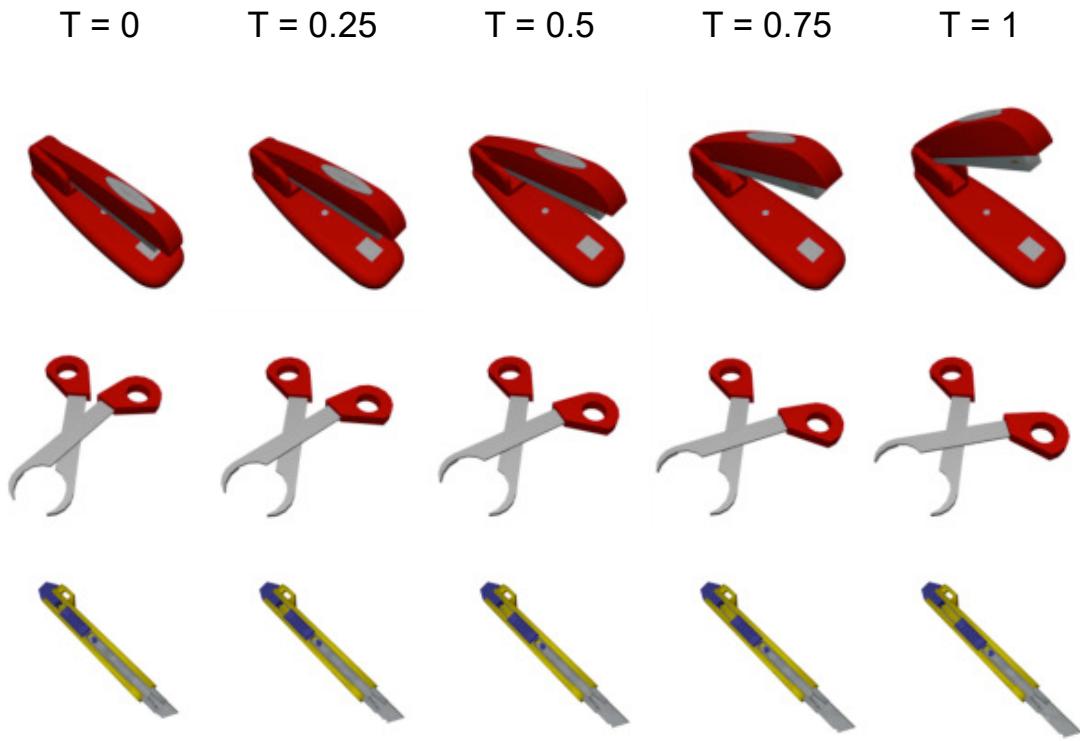


Figure 4: Qualitative Results on objects with different joints and distinct geometry structures.

undergoing continuous motion, with smooth transitions between configurations. These intermediate frames demonstrate that Part²GS produces consistent motion paths through the full articulation sequence. Figure 5 highlights the quality of Part²GS part discovery. Across multiple object categories and motion types, Part²GS achieves cleaner segmentation boundaries and better part consistency across start, end, and canonical states. Notably, Part²GS avoids over-segmentation in symmetric or ambiguous regions, e.g., drawers or doors that move similarly, while methods without explicit part supervision often produce inconsistent groupings.

D. Broader Impact

The ability to accurately reconstruct and articulate 3D objects has far-reaching implications across robotics, simulation, and digital twin technologies. Part²GS contributes to this space by enabling precise, physically grounded modeling of complex articulated objects from visual observations. This can facilitate improved interaction and manipulation in embodied agents, enhance simulation fidelity in virtual environments, and support scalable generation of articulated assets for industrial and educational applications. For example, in digital content creation, Part²GS can lower the barrier to creating accurate, controllable 3D assets, aiding designers and animators who rely on physically plausible models. Moreover, the framework could serve as a foundation for affordance learning, enabling agents to infer function from form. While the ability to digitize and manipulate real-world objects raises potential concerns around privacy, intellectual property, or misuse in synthetic media, our model is designed for research and educational use. We encourage responsible deployment practices aligned with consent and attribution norms. Compared to large-scale generative systems, our model is computationally lightweight and environmentally efficient, and we view its benefits in controllable, interpretable object modeling as outweighing its risks when applied ethically.

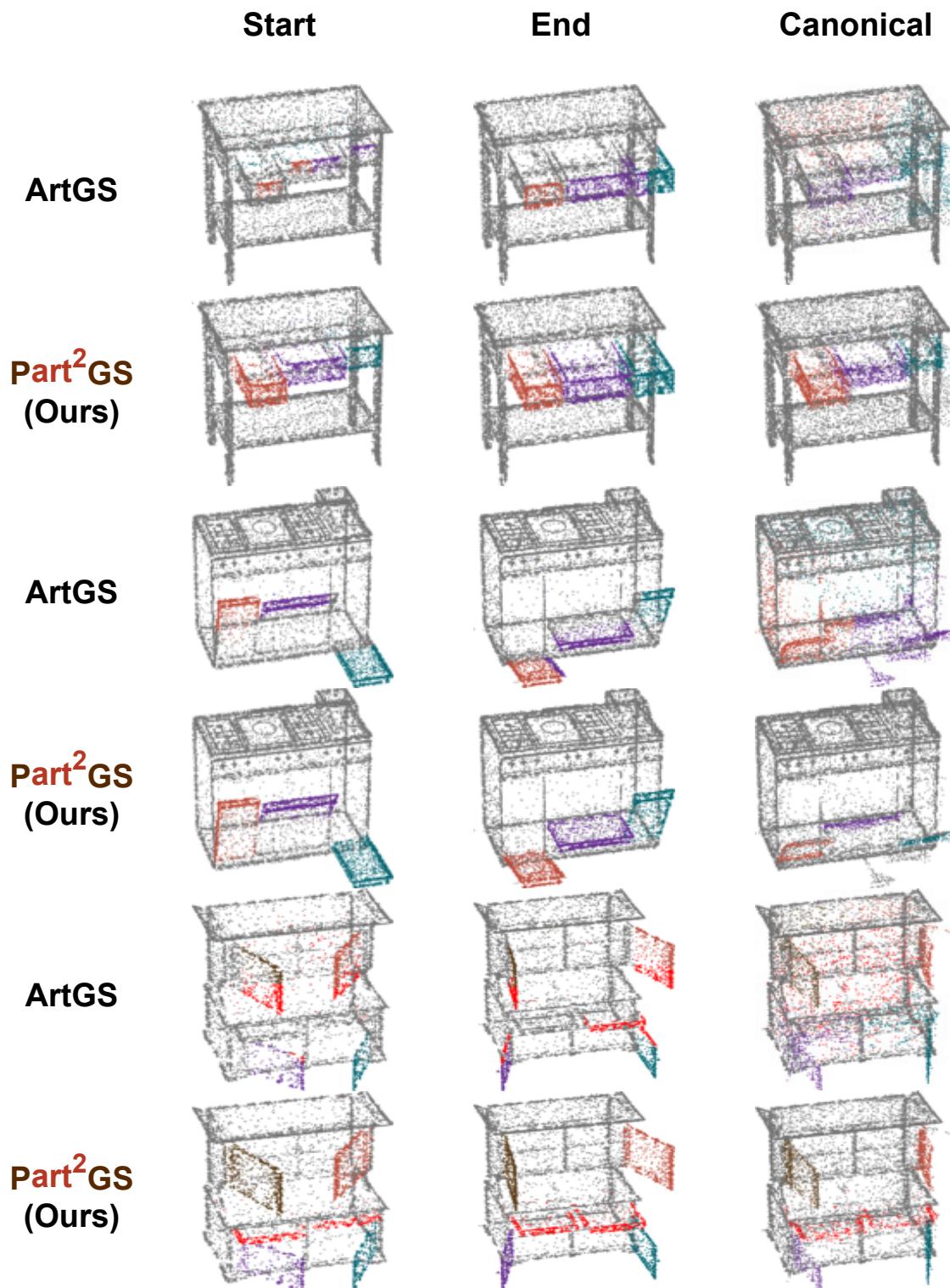


Figure 5: Qualitative Results on part discovery across object states.

E. Limitations

Our work introduces Part²GS, a physically grounded, part-aware framework for articulated 3D reconstruction that bridges the gap between neural point-based modeling and rigid-body articulation. We demonstrate that, by integrating motion-informed part discovery, SE(3) transform recovery, and differentiable physical constraints, Part²GS achieves accurate and interpretable reconstructions of complex objects with minimal supervision (*e.g.*, 3× better than the previous state-of-the-art). Despite these strengths, our method inherits several limitations that point toward promising future directions.

One limitation lies in its reliance on paired observations across two articulation states. While this setup enables robust part discovery and motion supervision, it assumes access to temporally aligned multi-view images of both joint configurations. In real-world scenarios, such clean and consistent state transitions may not be available, especially in unconstrained videos or partial scans.

Extending the framework to operate under weaker temporal or view constraints, possibly by integrating learned priors over articulation trajectories or leveraging video cues, remains an important direction. Another failure mode arises when distinct object parts undergo nearly identical transformations across observed states, *e.g.*, symmetric drawers being pulled out in unison. Since our articulation model leverages motion-informed priors and per-part SE(3) estimation, it assumes discriminative displacement trajectories to infer part-specific transformations. In degenerate cases with high motion similarity, Part²GS may fail to disentangle the parts, leading to collapsed representations or joint under-segmentation. This limitation highlights the need for more robust cues beyond geometric displacement alone, such as contact forces or kinematic priors, to resolve ambiguity. Addressing such scenarios will be critical for extending Part²GS to broader tasks like affordance-based manipulation or learning from video with subtle motion differences.