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

REArtGS++: Generalizable Articulation Reconstruction with Temporal Geometry Constraint via Planar Gaussian Splatting

Di Wu[1,2,4*], Liu Liu [3,5,6†], Anran Huang[3], Yuyan Liu[3], Qiaojun Yu[7], Shaofan Liu[3],
Liangtu Song[1] , Cewu Lu[7]
1 *Hefei Institutes of Physical Science, Chinese Academy of Sciences*
2 *University of Science and Technology of China*
3 *Hefei University of Technology* 4 *RoboticsX, Tencent*
5 *Jianghuai Advance Technology Center*
6 *Anhui Provincial Key Laboratory of Humanoid Robots*
7 *Shanghai Jiao Tong University*
Email: wdcs@mail.ustc.edu.cn, liuliu@hfut.edu.cn

Abstract

Articulated objects are pervasive in daily environments, such as drawers and refrigerators. Towards their part-level surface reconstruction and joint parameter estimation, REArtGS [25] introduces a category-agnostic approach using multi-view RGB images at two different states. However, we observe that REArtGS still struggles with screw-joint or multi-part objects and lacks geometric constraints for unseen states. In this paper, we propose REArtGS++, a novel method towards generalizable articulated object reconstruction with temporal geometry constraint and planar Gaussian splatting. We first model a decoupled screw motion for each joint without type prior, and jointly optimize part-aware Gaussians with joint parameters through part motion blending. To introduce time-continuous geometric constraint for articulated modeling, we encourage Gaussians to be planar and propose a temporally consistent regularization between planar normal and depth through Taylor first-order expansion. Extensive experiments on both synthetic and real-world articulated objects demonstrate our superiority in generalizable part-level surface reconstruction and joint parameter estimation, compared to existing approaches. Project Site: https://sites.google.com/view/reartgs2/home.

1. Introduction

Articulated objects are ubiquitous in daily life, such as drawers and laptops, and their surface reconstruction holds significant value across various fields, including robotics and embodied intelligence [22, 32, 33]. Unlike static objects, articulated objects consist of multiple parts with diverse kinematic structures, requiring additional part-level mesh reconstruction and joint parameter estimation. Under this circumstance, ASDF [16] and DITTO [8] train generative models of articulated objects using expensive 3D supervisions. However, these methods exhibit limited generalization to unseen categories due to the variations in shape and motion of articulated objects. To address this limitation, PARIS [12] achieves category-agnostic articulated modeling, i.e., part-level reconstruction and motion analysis, using neural implicit radiance fields, only with multi-view RGB images from two different states. Nevertheless, PARIS faces challenges in multi-part perception and smooth dynamic surface reconstruction, suffering from the entangled implicit representation.

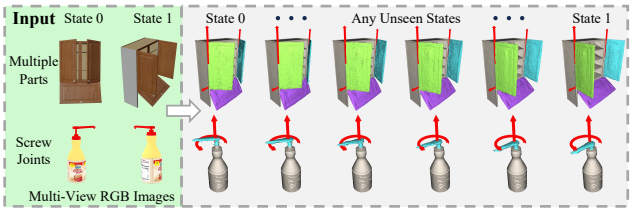


Figure 1. Given multi-view RGB images at arbitrary two different states of an unseen articulated objects, our method achieves high-quality part-level dynamic reconstruction and joint parameter estimation, without any external models.

Recently, ArtGS [13] and REArtGS [25] achieve efficient and realistic articulated object reconstruction and joint parameter estimation from two-state RGB-D and RGB images respectively, leveraging the explicit representation of

*Work done as an intern at Tencent RoboticsX
†Corresponding author

3D Gaussian Splatting (3DGS) [9]. However, we observe that these methods still suffer from the following limitations: (1) They rely on joint type priors, which require additional pipelines and pre-defined thresholds to obtain, and the joint types are assumed to be either rotation or translation. It may lead to significant errors when dealing with screw-joint and multi-part objects; (2) It is difficult to impose temporal geometric constraints from only two-state supervisions. Since the joint parameters and Gaussian primitives are jointly optimized, lacking geometric constraint for unseen states futher affects joint parameter estimation. Although REArtGS enhances geometric constraints by incorporating signed distance fields (SDF), it fails to perform temporal regularization due to the static mapping of SDF.

To address the aforementioned limitations, we propose REArtGS++, a novel framework of **R**econstructing g**E**neralizable **Art**iculated objects via planar **G**aussian **S**platting and temporal geometric constraint, shown as Fig. 1. We model a decoupled screw motion for each joint and derive Gaussian's motion from part motion blending, taking part segmentation probability as blending weights. In this way, we jointly optimize part-aware Gaussians and joint parameters without joint type priors.

To introduce dynamic geometric constraint for improved optimization of articulated modeling, we first compress the 3D Gaussians to flat planes to obtain accurate normals and unbiased depth maps following PGSR [1]. Then we propose a temporal geometric constraint, extending the consistent regularization of normal and depth from discrete states to entire time interval through Taylor first-order expansion. Moreover, we propose local consistent voting to address the ambiguous segmentation of overlapping regions. Experimental results of both synthetic and real-world data demonstrate our superiority over existing state-of-the-art (SOTA) methods.

Our main contributions can be summarized as: (1) We model a part-level blending screw motion, achieving unsupervised part segmentation and arbitrary joint parameter estimation, without joint type priors; (2) We propose a temporal geometry constraint for the entire motion process in an unsupervised manner, through Taylor first-order expansion. (3) Extensive experimental results prove that our method is superior to existing SOTA methods in both synthetic and real data.

2. Related Works

2.1. Dynamic Reconstruction and Segmentation via 3DGS

3D Gaussian Spaltting [9] and its subsequent works [1, 5, 15, 31] achieve realistic and efficient scene reconstruction via explicit 3D Gaussian primitives. Recent studies [30] have increasingly explored the potential of 3D Gaussian Splatting (3DGS) for dynamic scene reconstruction. Specifically, Deformable-3DGS [9] and 4DGS [27] leverage deformation networks alongside 3D Gaussians, obtaining time-independent Gaussians for dynamic scenes. SC-GS [6] and SP-GS [2] employ superpoints for dynamic Gaussian modeling and control.

Several works attempt to perform instance segmentation with 3D Gaussians [7, 17, 19, 20, 35]. These methods achieve 3D Gaussian segmentation by introducing 2D masks obtained from foundation models such as SAM [10] or DINO [34]. However, they rely on the prior knowledge provided by external models and typically struggle with part segmentation.

2.2. Articulated Object Reconstruction and Motion Analysis

High-quality object reconstruction remains a cornerstone of computer vision, attracting intensified scrutiny within the research community [14, 24, 26]. For articulated objects, previous methods [16, 21] only focus on their dynamic surface reconstruction. To achieve both surface reconstruction and motion analysis for articulated objects, PARIS [12, 23] employ composite neural radiance fields to jointly optimize geometry and motion through multi-view RGB images of two different states. Following this setting, several methods [3, 13, 18, 25] introduce 3DGS to articulated object reconstruction and joint parameter estimation. However, these methods require additional pipelines to obtain joint type priors, which are limited to rotation or translation, and affect the performance when dealing with screw-joint or multi-part objects.

3. Preliminary: Planar Gaussian Spaltting

3DGS [9] uses a set of Gaussians to explicitly represent the scene instead of neural implicit representation. The rendering color can be derived from the α-blending as follows:

$$C = \sum_{i=1}^{N_{\mathcal{G}}} \mathbf{c}_i \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j) \tag{1}$$

where $N_{\mathcal{G}}$ is the number of ordered points overlapping the pixel, $\mathbf{c}_i$ is the color described by Spherical Harmonics, and α_i is obtained from Gaussian's rendering contribution multiplied with a learnable opacity coefficient $\sigma \in (0, 1)$.

3DGS learns the 3D Gaussians through a rendering loss $\mathcal{L}_{\text{render}}$, which can be represented as:

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

where $\mathcal{L}_1$ is a pixel-level L1 norm and $\mathcal{L}_{\text{D-SSIM}}$ is the D-SSIM loss [9].

Recently, PGSR [1] obtains accurate normals and unbiased depth estimates by compressing 3D Gaussians to 2D

planar representations, enhancing geometric learning. Concretely, the normal maps $\mathbf{N}$ of planar Gaussians can be naturally rendered via their shortest axes and view directions, and the unbiased depth can be estimated as:

$$\mathbf{D}(\rho) = \frac{\mathbf{d}}{\mathbf{N}(\rho)\mathbf{K}^{-1}\widetilde{\rho}} \tag{3}$$

where $\mathbf{d}$ is the distance map from Gaussian's plane to camera center and $\mathbf{K}$ denotes the camera intrinsic matrix. p and $\widetilde{\rho}$ are 2D position and its homogeneous coordinate representation on image plane respectively. The rendered depth is derived from the intersection point of rays and Gaussian flat planes, matching the actual surface without deviation.

4. Methodology

Our overall framework is shown in Fig. 2. Given an unseen articulated object with k parts, our goal is to achieve part-level surface mesh reconstruction and joint parameter estimation, using multi-view posed RGB images $\mathbf{I}_0, \mathbf{I}_1$ in arbitrary two different states without depth supervision. Then we can generate dynamic surface mesh at any state $t = [0, 1]$. The parameters of each joint include rotation angle $\theta \in \mathbb{R}$, rotation axis $\mathbf{a} \in \mathbb{R}^3$, pivot point $o \in \mathbb{R}^3$, and translation $\mathbf{t} \in \mathbb{R}^3$. Note that the joint may involve any rigid motion (rotation, translation and screw), and we do not need to determine their types before optimization. The detailed pipelines are elaborated below.

4.1. Part-Aware Planar Gaussian Representation

We first introduce a scale loss $\mathcal{L}_{\text{scale}}$ to flatten 3D Gaussian primitives into 2D planes.

$$\mathcal{L}_{\text{scale}} = \frac{1}{N_{\mathcal{G}}} \sum_{i}^{N_{\mathcal{G}}} \|\min(s_1, s_2, s_3)\| \tag{4}$$

where s_1, s_2, s_3 are the components of Gaussian's scale $\mathbf{s}$ along each direction. In this way, we can obtain accurate normals and unbiased depth maps through Eq. 3, facilitating the temporal geometric constraint in Sec. 4.3.

Subsequently, we introduce a segmentation mask $\mathbf{M} = \{m_1, m_2, ..., m_k\}$ for each planar Gaussian, representing the probability belonging to each part. Intuitively, the probability that a Gaussian primitive $\mathcal{G}_i$ belongs to a part is inversely proportional to its distance from the part's center. Inspired by REACTO [21], we first denote part centers as learnable parameters $(\mathbf{O}, \mathbf{V}, \mathbf{\Lambda})$, where $\mathbf{O} \in \mathbb{R}^{k\times3}$, $\mathbf{V} \in \mathbb{R}^{k\times3\times3}$, $\mathbf{\Lambda} \in \mathbb{R}^{k\times3}$ are part centers, center orientations and scale vectors respectively. For Gaussian $\mathcal{G}_i$, the relative distances $\mathbf{L}_i$ between Gaussian center μ_i and part centers are defined as: $\mathbf{L}_i = \mathbf{V}(\mu_i - \mathbf{O})\mathbf{\Lambda}$. Then the Mahalanobis distances γ can be derived as: $\gamma_i = \mathbf{L}_i^T\mathbf{L}_i$. We calculate part segmentation mask $\mathbf{M}_i$ of $\mathcal{G}_i$ through the Mahalanobis distances and a residual term learned by an multilayer perceptron (MLP), which is illustrated in the supplementary materials.

4.2. Part-level Blending Motion

First, we briefly analyze the limitations of existing methods in modeling Gaussian motion. REArtGS models Gaussian motion as time interpolation of rotation or translation. ArtGS learns Gaussian motion through dual quaternions, but lacks explicit modeling for the pivot of rotation joints, causing an entangled representation of the pivot and translation. We provide a detailed proof in the supplement. Consequently, these methods struggle with estimating screw motion and cause additional errors when pre-determining joint types for multi-part objects.

To model a screw motion SE (3) without prior, we decouple the parameters ω of each joint into rotation and translation and set $\omega = \{\mathbf{q}(\theta, \mathbf{a}), \mathbf{o}, \mathbf{t}\}$ as trainable parameters for each part. For simplicity, we use $\mathbf{q}$ to represent joint quaternion $\mathbf{q}(\theta, \mathbf{a})$ in the following. Then the motion of planar Gaussian $\mathcal{G}_i$ can be derived from the blending of part motions, where the segmentation probability $\{m_1^i, m_2^i, ..., m_k^i\}$ serves as the blending weights. We formulate the transformed position $\mu_i(t)$ as following:

$$\mu_i(t) = \sum_{j=1}^{k} m_j \left[R_j(\mathbf{q}(t))(\mu_i - \mathbf{o}_j) + \mathbf{o}_j + \mathbf{t}_j(t)\right] \tag{5}$$

where $t \in [0, 1]$ denotes the state, and $R(\mathbf{q}(t))$ and $\mathbf{t}(t)$ is the time-varying rotation matrix and translation. We illustrate their calculation in detail below.

We define the canonical state as $t^* = 0.5$ to ensure the rotation angle θ belongs to $[-\pi/2, \pi/2]$, preventing the singularity in exponential coordinates when $\|\theta\| > \pi$, following REArtGS. The rotation angle and translation at state t can be represented as:

$$\theta(t) = \frac{(t - t^*)}{t^*}\theta, \quad \mathbf{t} = \frac{(t - t^*)}{t^*}\mathbf{t} \tag{6}$$

and the time-varying rotation quaternion can be formulated as:

$$\mathbf{q}(t) = \cos(\frac{\theta(t)}{2}) + \mathbf{a}\sin(\frac{\theta(t)}{2}) \tag{7}$$

Hence, the rotation matrix $R(\mathbf{q}(t))$ can be derived through $\mathbf{q}(t)$. In this way, we jointly optimize the part segmentation $\mathbf{M}$ and joint parameters in a unsupervised manner.

4.3. Temporal Geometric Constraint

To enhance the geometric constraint in dynamic rendering, we propose an unsupervised consistency regularization between depth and normals of Gaussians. Leveraging the joint parameters ω, we can obtain the normal map

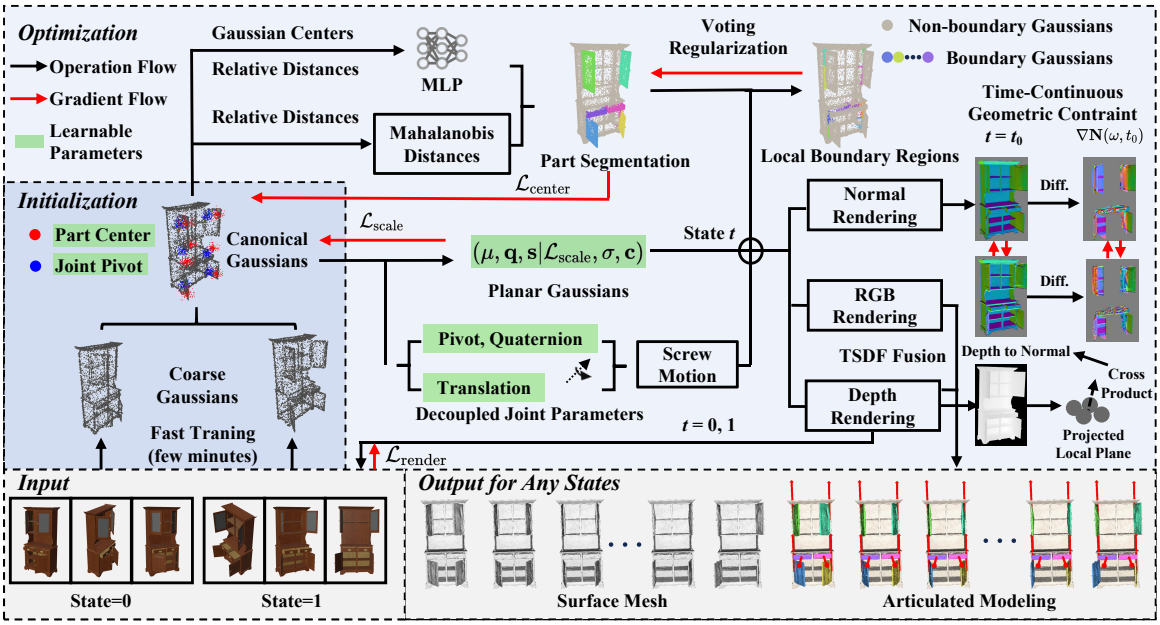


Figure 2. Framework of REArtGS++. Our method jointly optimizes part segmentation, joint parameters and planar Gaussians using multi-view RGB images from arbitrary two states, and achieves high-quality part-level mesh reconstruction of any states and accurate joint parameter estimation for an unseen articulated object. "Diff." denotes the difference approximation.

Table 1. **Quantitative results on PARIS dataset.** We implement all methods without depth supervision for fair comparison. Axis Pos results are measured by mm. "-" indicates the object containing only prismatic joints. We highlight best and second best results.

Metric	Method	Synthetic Objects											Real Objects		
		FoldChair	Fridge	Laptop	Oven	Scissor	Stapler	USB	Washer	Blade	Storage	Mean	Fridge	Storage	Mean
Axis Ang	PARIS [12]	15.72	2.02	0.03	7.43	16.62	7.17	0.71	15.40	38.28	0.03	10.34	2.90	20.16	11.53
	DTA [23]	0.03	0.08	0.07	0.21	0.10	0.08	0.11	0.38	0.18	0.09	0.13	2.26	10.57	6.42
	ArtGS [13]	0.01	0.03	**<0.01**	0.02	**<0.01**	0.01	**0.02**	0.04	0.02	0.01	0.02	2.03	4.56	3.30
	REArtGS [25]	0.01	0.02	0.02	0.07	0.05	0.06	0.04	0.08	0.04	0.04	0.04	2.19	87.51	44.85
	REArtGS++ (Ours)	**<0.01**	**0.01**	**<0.01**	**<0.01**	0.04	**<0.01**	0.04	**0.02**	**0.01**	**<0.01**	**0.01**	**1.92**	**3.34**	**2.63**
Axis Pos	PARIS [12]	10.30	9.18	0.76	5.15	15.31	35.07	13.26	24.14	-	-	14.15	50.28	-	50.28
	DTA [23]	1.47	1.16	2.32	1.18	1.95	1.67	1.46	4.87	-	-	2.01	59.13	-	59.13
	ArtGS [13]	**0.06**	0.43	0.94	1.35	0.18	**0.38**	0.15	1.06	-	-	0.51	42.83	-	42.83
	REArtGS [25]	6.24	**0.15**	1.31	3.64	0.06	0.72	1.56	**0.04**	-	-	1.72	45.72	-	45.72
	REArtGS++ (Ours)	0.37	0.34	**0.73**	**0.71**	**<0.01**	0.49	**0.04**	0.95	-	-	**0.45**	**42.66**	-	**42.66**
Part Motion	PARIS [12]	45.80	68.21	0.03	9.70	71.42	128.73	34.57	43.58	0.45	0.30	40.28	2.23	0.57	1.40
	DTA [23]	0.10	0.12	0.11	0.12	0.34	0.08	0.15	0.24	**<0.01**	**<0.01**	0.13	2.56	0.18	1.37
	ArtGS [13]	0.03	0.03	0.03	0.01	**0.01**	0.03	**<0.01**	0.02	**<0.01**	0.01	0.02	2.27	0.15	1.21
	REArtGS [25]	0.31	0.23	0.06	0.08	0.06	0.15	0.05	0.04	**<0.01**	0.08	0.11	38.38	0.52	19.45
	REArtGS++ (Ours)	**0.02**	**<0.01**	**0.02**	**<0.01**	0.03	**0.01**	0.03	**0.01**	**<0.01**	**<0.01**	**0.01**	**2.16**	**0.13**	**1.15**
CD-s	PARIS [12]	8.51	6.30	0.25	4.57	12.71	2.85	1.95	**8.37**	10.64	8.62	6.48	10.74	27.61	19.18
	DTA [23]	0.27	2.48	0.64	10.06	1.64	3.08	2.64	20.73	0.46	10.73	5.74	4.50	30.98	17.74
	ArtGS [13]	**0.16**	1.89	0.80	8.58	0.50	3.55	2.51	23.63	0.51	11.82	5.40	3.39	18.19	10.79
	REArtGS [25]	0.31	1.53	**0.48**	6.59	**0.25**	**1.47**	**0.82**	16.04	0.69	**7.47**	3.57	7.67	22.53	15.10
	REArtGS++ (Ours)	0.27	**0.97**	0.53	**4.14**	0.37	3.60	3.12	10.59	**0.39**	10.41	**3.44**	**1.33**	**10.98**	**6.16**
CD-m	PARIS [12]	4.05	1.26	**0.15**	3.70	10.06	1.18	3.02	1.52	5.13	20.67	5.07	74.20	314.37	194.29
	DTA [23]	0.46	2.27	0.96	1.36	10.25	1.61	1.37	4.45	2.07	5.34	3.01	3.63	81.36	42.50
	ArtGS [13]	0.52	1.40	0.98	2.52	0.70	0.86	1.15	4.67	**1.23**	6.56	2.06	3.23	33.65	18.44
	REArtGS [25]	**0.30**	**0.66**	0.71	1.25	**0.25**	**0.21**	**0.40**	**0.42**	1.97	12.39	1.85	2.87	175.78	89.33
	REArtGS++ (Ours)	0.77	1.25	0.60	**0.85**	0.83	0.90	1.15	0.89	2.54	**4.78**	**1.46**	**1.75**	**13.85**	**7.80**
CD-w	PARIS [12]	8.27	2.83	3.13	11.08	11.68	2.80	12.46	19.67	0.81	8.37	8.11	8.82	24.73	16.78
	DTA [23]	0.58	2.36	0.75	9.35	0.65	2.95	1.43	18.68	0.51	8.34	4.56	4.47	23.48	13.98
	ArtGS [13]	**0.44**	2.01	0.96	8.32	0.72	**2.71**	1.57	21.69	0.67	8.69	4.78	3.26	12.26	15.52
	REArtGS [25]	0.70	1.64	0.65	10.89	**0.31**	3.26	1.54	17.82	**0.40**	7.71	4.49	2.93	11.48	7.21
	REArtGS++(Ours)	0.51	**1.20**	**0.48**	**3.72**	0.56	2.87	**1.30**	**13.97**	0.50	**7.39**	**3.25**	**1.52**	**10.63**	**6.08**

$\mathbf{N}(\omega, t)$ through α-blending similar to Eq. 1, and depth map $\mathbf{D}(\omega, t)$ via Eq. 3 at any state $t \in [0, 1]$. Inspired by PGSR,

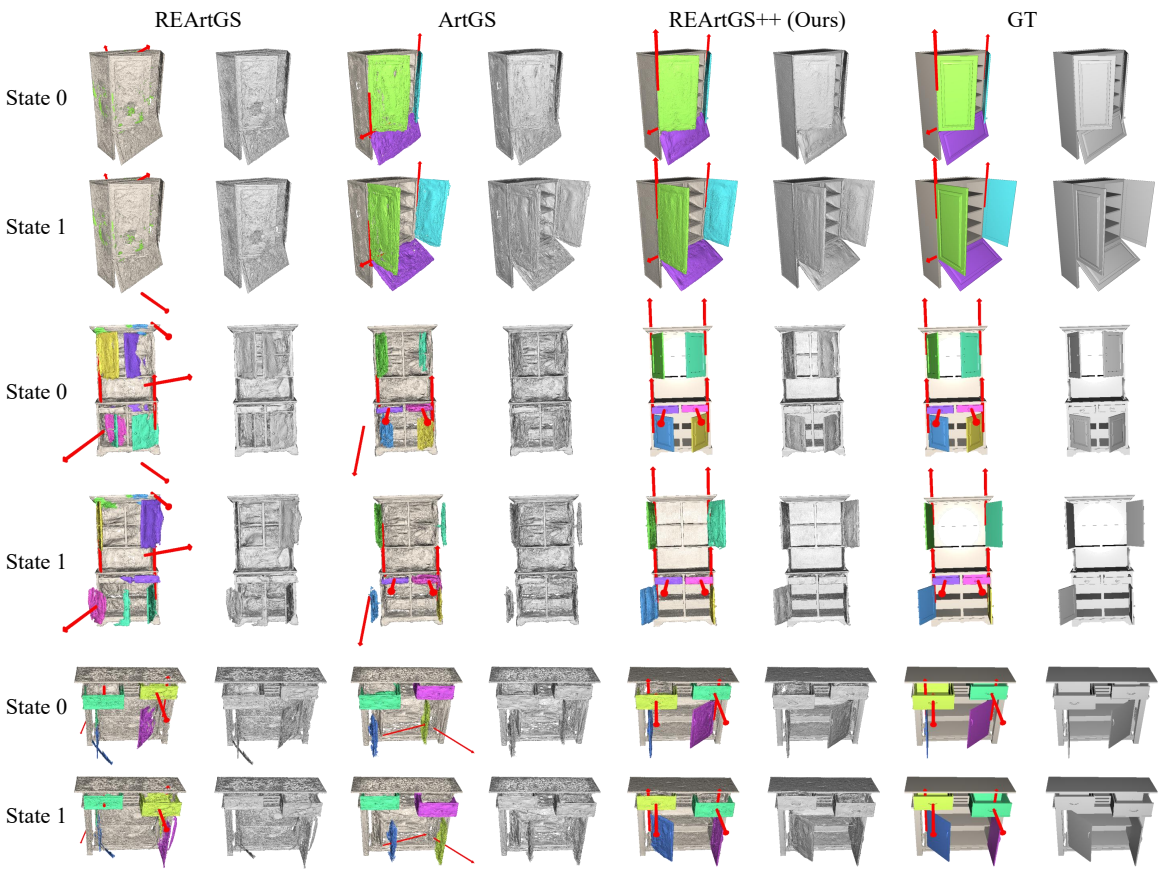


Figure 3. The qualitative results of dynamic surface reconstruction at start state and end state on ArtGS-Multi dataset. We show both part segmentation and surface meshes for best comparison. The red arrows represent joints.

we take the Gaussian as a local plane and sample a local patch (4 neighboring pixels) from $\mathbf{D}(\omega, t)$ and project them into 3D points. The estimated normal can be derived from their cross product, and we acquire estimated normal map $\bar{\mathbf{N}}(\omega, t)$ by repeating the process for entire depth map. A natural technique is to align $\bar{\mathbf{N}}$ with $\mathbf{N}$, as in PGSR. However, this approach is limited to a single discrete state and cannot regularize the entire interval $t \in [0, 1]$. To solve the above problem, we adopt the Taylor first-order expansion to approximate $\mathbf{N}(\omega, t)$, which can be formulated as:

$$\mathbf{N}(\omega, t) \approx \mathbf{N}(\omega, t_0) + \lim_{t \to t_0} \frac{\mathrm{d}N(\omega, t)}{\mathrm{d}t}(t - t_0) \tag{8}$$

where $t_0 = 0, 1$ to ensure that the depth and normal renderings can be obtained simultaneously through a single forward when calculating color rendering loss. Furthermore, we use a difference approximation to represent the gradient of the normal against time, shown as:

$$\nabla\mathbf{N}(\omega, t_0) = \lim_{t \to t_0} \frac{\mathrm{d}N(\omega, t)}{\mathrm{d}t} \approx \frac{N(\omega, t) - N(\omega, t^*)}{t - t^*} \tag{9}$$

Since the Gaussians do not apply motion transformation at canonical state t^*, this approximation reduces computational costs and prevents the accumulation errors of motion during optimization. Note that $N(\omega, t^*)$ does not participate in back-propagation to save memory. The same approximation is adopted for $\bar{\mathbf{N}}(t)$. In this way, the time-continuous geometry constraint for the entire interval $t \in [0, 1]$ can be represented as following:

$$\begin{aligned}\mathcal{L}_{geo} = (1 - \nabla\mathbf{I}(t_0)) \big(&\|\bar{\mathbf{N}}(\omega, t_0) - \mathbf{N}(\omega, t_0)\| \\ &+ \|\nabla\bar{\mathbf{N}}(\omega, t_0) - \nabla\mathbf{N}(\omega, t_0)\|\big)\end{aligned} \tag{10}$$

where $\nabla\mathbf{I}(t)$ is the image gradient to avoid computations at the edges of images.

4.4. Local Consistent Voting

We observe the distance-based segmentation $\mathbf{M}$ often fails to learn a probability distribution with clear tendencies at boundary Gaussians $\hat{\mathcal{G}}$ between adjacent parts, leading to ambiguous and inconsistent part segmentation in such regions. To address this, we introduce a local consistent voting mechanism to constrain the consistency of part segmentation for boundary Gaussians $\hat{\mathcal{G}}$.

Given a part j, we define a Gaussian $\mathcal{G}_i$ within it as a

boundary Gaussian when meeting the following condition:

$$\sum \frac{\Gamma\left(\max(\mathbf{M}_{\Omega(\mathcal{G}_i)}) \neq m_j\right)}{|\Omega(\mathcal{G}_i)|} \geq \beta \tag{11}$$

where $\Omega(\mathcal{G}_i)$ is the neighbor Gaussians of $\mathcal{G}_i$ sampled by K-Nearest Neighbor (KNN) and m_j denotes the probability belonging to part j. Γ is an indicator function and β is set to 0.2.

The boundary Gaussians are then divided into $4 \times k$ local regions by k-means algorithm. For each region, we derive a voting segmentation probability $\mathbf{M}_{\text{vote}}$ through calculating the probability distributions $\mathbf{M}$ of all points, weighted by their distance to regional center. We formulate the calculation as following:

$$\mathbf{M}_{\text{vote}} = \sum_{i}^{\mathcal{N}(\mathcal{G}_i)} \frac{\Phi_{\text{softmax}}(-\delta) M_i}{\sum_i^{\mathcal{N}(\mathcal{G}_i)} M_i} \tag{12}$$

where Φ_{softmax} is the Softmax activation and δ represents the distance from Gaussian $\mathcal{G}_i$ to regional center.

Finally, we constrain the local segmentation consistency by minimizing the divergence between learned distribution for a boundary Gaussian and the voting distribution, which can be represented as:

$$\mathcal{L}_{\text{vote}} = \sum_{n}^{|\mathcal{N}|} \sum_{\mathcal{G}_i \in \mathcal{N}_n(\mathcal{G}_i)} \frac{1}{|\mathcal{N}_n(\mathcal{G}_i)|} D_{\text{KL}}\left(M_{\text{vote}} \,\|\, M_i\right) \tag{13}$$

where $D_{\text{KL}}(\cdot\|\cdot)$ denotes the KL divergence and $|\mathcal{N}|$ is the number of local regions. Intuitively, the regularization introduces spatial context information for part segmentation by aggregating probability distributions of local regions, instead of relying solely on point-wise understanding.

4.5. Initialization and Optimization

Inspired by ArtGS, we first perform a fast optimization for two sets of Gaussians at both start and end states using vanilla 3DGS pipeline [9], then identify the dynamic Gaussians for each state by criterion: $\Delta\mathbf{x}_i > \tau\mathbf{x}$, where $\Delta\mathbf{x}_i$ is the Chamfer Distance (CD) from Gaussian $\mathcal{G}_i$ at one state to all Gaussians of another state. $\mathbf{x}$ is the maximum CD and τ is set to 0.02. We initialize static and dynamic Gaussians of canonical state as the mean of the two states, using Hungarian matching.

Taking the static Gaussian as a cluster, we adopt K-means clustering algorithm to divide dynamic Gaussians into $k-1$ clusters and set cluster centers as the initialization of k-part centers. We set the initial scale of each part as the maximum distance from the farthest point to the cluster center. Given a cluster of part j, we extract the Gaussians of the connected regions between the static part and part j through a similar mechanism in Sec. 4.4. Then the mean position of the Gaussians within a connected region serves as initial pivot point for corresponding joint.

Once the initialization is completed, we optimize the dynamic rendering of planar Gaussians through the multi-view images at two states. The rendering loss $\mathcal{L}_{\text{render}}$ can be calculated through Eq. 2 using $\mathbf{I}_t$ and $\bar{\mathbf{I}}(\omega, t)$, where $\bar{\mathbf{I}}$ denotes the rendered images and $t = 0, 1$. In this way, our method also supports multi-state observation inputs $\{\mathbf{I}_0, \mathbf{I}_{t_1}, \mathbf{I}_{t_2}, ..., \mathbf{I}_1\}$ by extending t to $\{0, t_1, t_2, ..., 1\}$. Please refer to the supplement for more details. Following ArtGS, we also introduce a center regularization $\mathcal{L}_{\text{center}}$ to align part centers with the mean positions of the corresponding Gaussians.

In summary, our total training objective can be represented as following:

$$\begin{aligned}\mathcal{L} = &\lambda_{\text{render}}\mathcal{L}_{\text{render}} + \lambda_{\text{scale}}\mathcal{L}_{\text{scale}} + \lambda_{\text{center}}\mathcal{L}_{\text{center}} \\ &+ \lambda_{\text{geo}}\mathcal{L}_{\text{geo}} + \lambda_{\text{vote}}\mathcal{L}_{\text{vote}}\end{aligned} \tag{14}$$

4.6. Dynamic Mesh Extraction

Once the optimization converges, we can extract dynamic surface meshes at any state $t \in [0, 1]$ via the planar Gaussians $\mathcal{G}$ and joint parameters $\{\omega_1, ..., \omega_k\}$. Specifically, we obtain the Gaussians $\mathcal{G}^j$ of a given part j by:

$$\mathcal{G}^j = \{\mathcal{G}_i | \max(\mathcal{G}_i = m_j), i \in N_{\mathcal{G}}\} \tag{15}$$

Then the dynamic Gaussians $\mathcal{G}^j$ at state t can be updated by Eq. 5. The RGB and depth renderings of each part can be directly obtained by Eq. 1 and Eq. 3. We employ the TSDF fusion algorithm to extract part-level dynamic meshes using these maps as inputs, where the voxel size is set to 0.04.

5. Experiment

5.1. Experimental Setting

Datasets. We compare our method with existing SOTAs on three datasets: (1) PARIS dataset [12], including 10 objects from PartNet-Mobility dataset [28]; (2) ArtGS-Multi [13], including 5 objects from PartNet-Mobility dataset, and We collect 2 additional screw joint objects from PartNet-Mobility dataset; (3) Real-world dataset, containing 2 objects from PARIS and 3 objects collected by ours. Each state provides 60-100 posed RGB images.

Metrics. Following ArtGS, we report the mean Chamfer Distance (CD, $\times 10^3$) on meshes for the whole surface (CD-w), static (CD-s), and dynamic parts (CD-m). Joint predictions are evaluated via angular error (**Axis Ang.**, in °) and pivot distance (**Axis Pos.**, in mm). Part motion is quantified by geodesic rotation (°) and Euclidean translation (m) errors. Lower (↓) is better for all metrics.

5.2. Experiments on Two-Part Articulated Objects

We present the quantitative and qualitative results on the PARIS dataset in Table. 1 and Fig. 4 respectively. More

Table 2. **Quantitative results on ArtGS-Multi dataset with additional 2 objects exhibiting screw joints**. Due to the large number of parts, we report the average metric for all movable parts. We implement all methods without depth supervision for fair comparison. Part motion error for screw objects is decomposed by translation|rotation. Axis Pos results are measured by mm. '-' indicates the object contains only prismatic joints. We highlight best and second best results.

Metrics	Method	Table 25493 (4 parts)	Table 31249 (5 parts)	Storage 45503 (4 parts)	Storage 47468 (7 parts)	Oven 101908 (4 parts)	Mean	Bottle 3398 (screw part)	Dispenser 103378 (screw part)	Mean
Axis Ang	ArtGS [13]	1.23	83.69	**0.01**	34.83	5.62	25.08	42.28	87.57	64.93
	REArtGS [25]	85.45	42.18	64.34	30.69	24.50	49.43	46.45	87.80	67.13
	REArtGS++ (Ours)	**0.02**	**0.05**	0.06	**2.58**	**0.03**	**0.55**	**0.01**	**0.12**	**0.07**
Axis Pos	ArtGS [13]	-	70.93	0.36	235.19	8.54	96.49	50.64	38.75	44.70
	REArtGS [25]	-	155.80	500.51	31.44	233.48	230.31	47.18	20.41	33.80
	REArtGS++ (Ours)	-	**0.10**	**0.21**	**0.06**	**1.62**	**0.50**	**1.04**	**4.03**	**2.54**
Part Motion	ArtGS [13]	**0.08**	22.50	**0.01**	18.75	5.73	9.41	0.06|65.87	0.02|31.84	0.04|48.86
	REArtGS [25]	0.23	13.09	46.09	25.41	55.53	28.07	0.04|63.74	0.03|45.90	0.04 |54.82
	REArtGS++ (Ours)	0.67	**0.09**	0.08	**0.05**	**0.21**	**0.22**	**0.01|0.02**	**0.01|0.44**	**0.01|0.23**
CD-s	ArtGS [13]	85.65	3.01	2.35	1.74	10.45	20.64	1.78	0.30	1.04
	REArtGS [25]	45.11	2.86	14.12	2.16	15.83	16.02	0.85	0.21	0.53
	REArtGS++ (Ours)	**0.77**	**1.26**	**1.35**	**0.61**	**5.03**	**1.80**	**0.64**	**0.18**	**0.41**
CD-m	ArtGS [13]	216.58	9.50	**0.39**	7.49	**0.79**	46.95	0.46	11.75	6.11
	REArtGS [25]	71.44	10.63	119.32	70.91	136.68	81.80	0.97	10.74	40.86
	REArtGS++ (Ours)	**1.39**	**4.89**	0.58	**3.19**	0.80	**2.17**	**0.08**	**5.70**	**2.89**
CD-w	ArtGS [13]	1.06	4.06	**2.29**	3.31	9.87	4.12	1.75	0.59	1.17
	REArtGS [25]	3.52	2.88	13.16	6.16	17.96	8.74	0.80	0.47	0.64
	REArtGS++ (Ours)	**0.72**	**1.13**	3.76	**2.13**	**5.14**	**2.58**	**0.59**	**0.31**	**0.45**

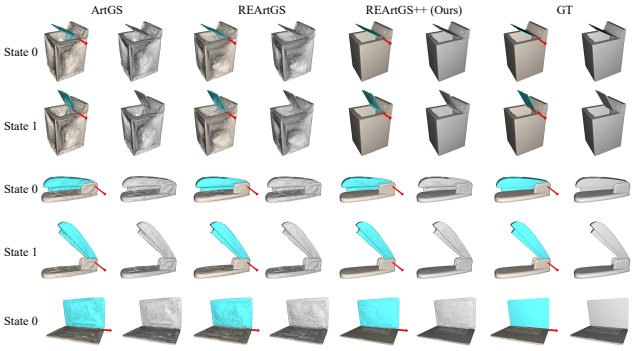


Figure 4. The qualitative results of dynamic surface reconstruction at start state and end state on PARIS dataset. We show both articulated modeling and surface meshes for best comparison.

qualitative results can be found in the supplementary material. Our method achieves the best results on the mean values of all metrics especially in CD-w of Synthetic objects, achieving a **27.6**% error reduction compared to REArtGS. This is primarily attributed to the enhancement in dynamic reconstruction quality through the dynamic geometric constraints. In addition, The superior performance of the joint parameter estimation results demonstrates the effectiveness of decoupling screw motion learning.

5.3. Experiments on Screw and Multi-Part Articulated Objects

The quantitative and qualitative results of screw and multi-part objects are provided in Table. 2. We also provide the

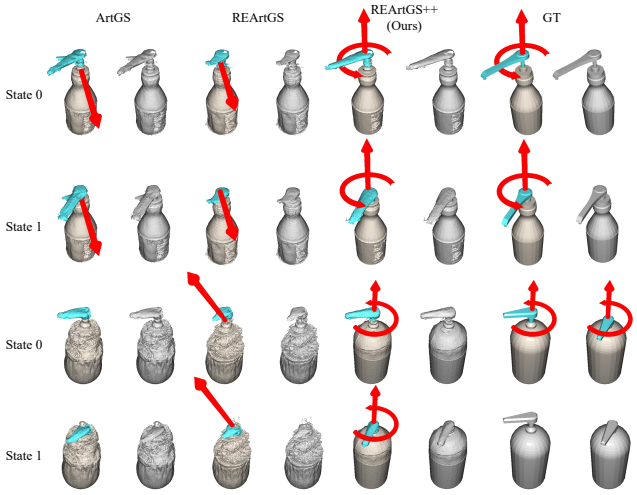


Figure 5. The qualitative results of dynamic surface reconstruction at start state and end state on PARIS dataset. We show both articulated modeling and surface meshes for best comparison.

qualitative results in Fig. 3 and Fig. 5 respectively. Our method exhibits significant advantages over existing methods in the mean values of all metrics, especially in Axis Ang, Axis Pos and Part Motion, achieving mean results of **0.55**, **0.50**, **0.22** respectively. ArtGS and REArtGS rely on prior knowledge of joint types, where incorrect priors severely degrade subsequent joint axis parameter estimation (e.g., *Table 31249* and *Storage 47468*), while our method does not require joint type estimation, exhibiting greater robustness for multi-part objects.

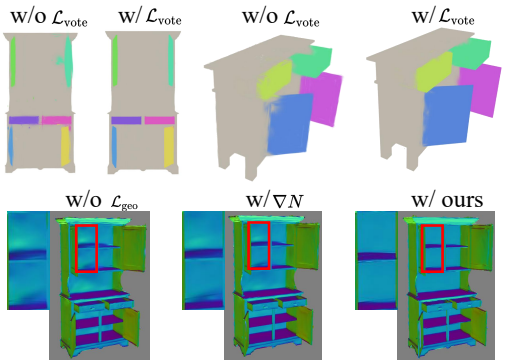


Figure 6. Qualitative results for the ablation of voting and geometric regularization. We use segmentation rendering results for intuitive visualization.

Table 3. **Ablation of key componets**. We report the average results on ArtGS-Multi and 2 screw-joint objects. Axis Pos results are measured by mm. "GS" and "dis" denote Gaussians and distance respectively.

Setting	Axis Ang	Axis Pos	Part Motion	CD-s	CD-m	CD-w
w/o planar GS	5.18	21.38	6.23	5.84	7.57	6.45
w/o screw motion	22.78	88.26	18.04	16.24	29.68	3.16
w/o MLP	20.72	34.01	27.52	14.38	76.50	2.72
w/o relative dis.	57.81	38.17	35.17	7.34	93.80	6.38
w/o initialization	33.47	71.74	40.13	6.81	126.03	8.43
w/o $\mathcal{L}_{\text{center}}$	28.48	47.36	12.63	3.65	31.54	3.32
w/o $\mathcal{L}_{\text{vote}}$	2.66	15.10	5.36	2.71	10.38	3.07
w/ all	**0.41**	**1.18**	**0.22**	**1.41**	**2.38**	**2.13**

5.4. Ablation Study

Ablation of key components. To further validate the effectiveness of our key components, we conducted ablation studies on ArtGS-Multi dataset with 2 additional objects exhibiting screw joints. We present the quantitative results in Table. 3 and a qualitative comparison of local voting regularization in Fig. 6. For "w/o planar GS", we employ the original 3DGS pipeline [9]; for "w/o screw motion", joint motions are represented with dual quaternions; for "w/o initialization", all part centers and joint pivots are randomly initialized. Experimental results demonstrate that each component contributes significantly to overall performance.

Ablation of temporal geometry constraints. To validate the effectiveness of the temporal geometric constraints, we conduct ablation studies on ArtGS-Multi dataset with 2 additional objects exhibiting screw joints. We present the quantitative results in Table. 4 and a qualitative comparison in Fig. 6. For "w/o ∇N", ∇N and $\nabla \bar{N}$ are ignored in Eq. 10; For "w/ random Δt" and "w/ tiny Δt", we use $t_0 \pm 0.1$ and a random $t_{\text{rand}} \in [0, 1]$ to replace t^* in Eq. 9. Experimental results demonstrate our approach in Sec. 4.3 exhibits clear advantage on all metrics, demonstrating temporal geometry constraint enhance the joint optimization of articulated modeling. Notably, "w/ random Δt" brings negative effects, as using other states for difference approximation introduces additional motion errors especially in early optimization.

Table 4. **Ablation of temporal geometry constraints**. We report the average results on ArtGS-Multi and 2 screw-joint objects.

Setting	Axis Ang	Axis Pos	Part Motion	CD-s	CD-m	CD-w
w/o $\mathcal{L}_{\text{geo}}$	4.04	16.08	4.73	4.42	5.70	5.06
w/o ∇N	3.61	9.52	3.04	3.86	3.72	4.41
w/ random Δt	5.83	23.10	6.24	6.60	10.07	6.34
w/ tiny Δt	2.65	11.84	3.74	4.78	3.58	4.55
w/ ours	**0.41**	**1.18**	**0.22**	**1.41**	**2.38**	**2.13**

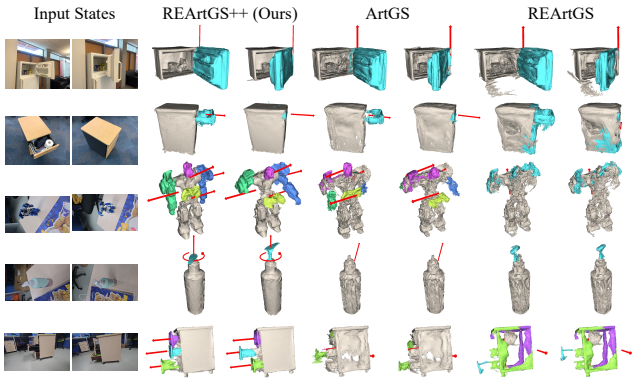


Figure 7. Qualitative results of part-level surface reconstruction at both start and end states on real-world objects.

5.5. Generalization to Real-World Articulated Objects

We present the quantitative results of 2 real-world objects from the PARIS dataset in Table. 1, and the qualitative results with additional 3 complex real objects we casually captured in Fig. 7. Experimental results prove that our method maintains superior articulated modeling results for multi-part and screw-joint objects in daily environments, demonstrating our generalization to the real world.

6. Conclusion and Limitations

In this paper, we propose REArtGS++, a novel method towards generalizable articulated object reconstruction via part-aware planar Gaussians. We model decoupled screw joint parameters to represent arbitrary motion and propose a time-continuous geometric regularization to enhance dynamic reconstruction. Both synthetic and real data experiments demonstrate our superior to previous SOTAs.

Our main limitations lie in transparent object reconstruction and the requirement for camera pose alignment between two states. Future work will combine depth estimation and transparent material modeling methods to address transparent surface perception [11, 29], and jointly refine two-state camera poses [4] in real world data.

Acknowledgements

This work was supported by National Natural Science Foundation of China under Grant 62302143.

References

[1] Danpeng Chen, Hai Li, Weicai Ye, Yifan Wang, Weijian Xie, Shangjin Zhai, Nan Wang, Haomin Liu, Hujun Bao, and Guofeng Zhang. Pgsr: Planar-based gaussian splatting for efficient and high-fidelity surface reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, 31 (9):6100–6111, 2025. 2

[2] Gang Zeng Diwen Wan, Ruijie Lu. Superpoint gaussian splatting for real-time high-fidelity dynamic scene reconstruction. In *Forty-first International Conference on Machine Learning*, 2024. 2

[3] 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. 2

[4] Sunghwan Hong, Jaewoo Jung, Heeseong Shin, Jisang Han, Jiaolong Yang, Chong Luo, and Seungryong Kim. Pf3plat: Pose-free feed-forward 3d gaussian splatting. *arXiv preprint arXiv:2410.22128*, 2024. 8

[5] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *SIGGRAPH 2024 Conference Papers*. Association for Computing Machinery, 2024. 2

[6] 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. *arXiv preprint arXiv:2312.14937*, 2023. 2

[7] Minchao Jiang, Shunyu Jia, Jiaming Gu, Xiaoyuan Lu, Guangming Zhu, Anqi Dong, and Liang Zhang. Votesplat: Hough voting gaussian splatting for 3d scene understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6456–6465, 2025. 2

[8] Zhenyu Jiang, Cheng-Chun Hsu, and Yuke Zhu. Ditto: Building digital twins of articulated objects from interaction. In *arXiv preprint arXiv:2202.08227*, 2022. 1

[9] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42 (4), 2023. 2, 6, 8

[10] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023. 2

[11] Mingwei Li, Pu Pang, Hehe Fan, Hua Huang, and Yi Yang. Tsgs: Improving gaussian splatting for transparent surface reconstruction via normal and de-lighting priors, 2025. 8

[12] Jiayi Liu, Ali Mahdavi-Amiri, and Manolis Savva. Paris: Part-level reconstruction and motion analysis for articulated objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 352–363, 2023. 1, 2, 4, 6

[13] 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 *The Thirteenth International Conference on Learning Representations*, 2025. 1, 2, 4, 6, 7

[14] Yuyan Liu, Li Zhang, Di Wu, Yan Zhang, Anran Huang, Zhi Wang, Liu Liu, and Dan Guo. Siam: Towards generalizable articulated object modeling via single robot-object interaction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(22):18478–18486, 2026. 2

[15] Tao Lu, Mulin Yu, Linning Xu, Yuanbo Xiangli, Limin Wang, Dahua Lin, and Bo Dai. Scaffold-gs: Structured 3d gaussians for view-adaptive rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20654–20664, 2024. 2

[16] Jiteng Mu, Weichao Qiu, Adam Kortylewski, Alan Yuille, Nuno Vasconcelos, and Xiaolong Wang. A-sdf: Learning disentangled signed distance functions for articulated shape representation. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12981–12991, 2021. 1, 2

[17] Minghan Qin, Wanhua Li, Jiawei Zhou, Haoqian Wang, and Hanspeter Pfister. Langsplat: 3d language gaussian splatting. *arXiv preprint arXiv:2312.16084*, 2023. 2

[18] Licheng Shen, Saining Zhang, Honghan Li, Peilin Yang, Zihao Huang, Zongzheng Zhang, and Hao Zhao. Gaussianart: Unified modeling of geometry and motion for articulated objects. *arXiv preprint arXiv:2508.14891*, 2025. 2

[19] Jin-Chuan Shi, Miao Wang, Hao-Bin Duan, and Shao-Hua Guan. Language embedded 3d gaussians for open-vocabulary scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5333–5343, 2024. 2

[20] Ola Shorinwa, Jiankai Sun, and Mac Schwager. Fast-splat: Fast, ambiguity-free semantics transfer in gaussian splatting. *arXiv preprint arXiv:2411.13753*, 2024. 2

[21] Chaoyue Song, Jiacheng Wei, Chuan Sheng Foo, Guosheng Lin, and Fayao Liu. Reacto: Reconstructing articulated objects from a single video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5384–5395, 2024. 2, 3

[22] Yuanfei Wang, Xiaojie Zhang, Ruihai Wu, Yu Li, Yan Shen, Mingdong Wu, Zhaofeng He, Yizhou Wang, and Hao Dong. Adamanip: Adaptive articulated object manipulation environments and policy learning. In *International Conference on Learning Representations*, 2025. 1

[23] 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 *CVPR*, 2024. 2, 4

[24] Di Wu, Linli Zhou, JinCheng Li, Jianqiao Xiong, and Liangtu Song. Explicit 3d reconstruction from images with dynamic graph learning and rendering-guided diffusion. *Neurocomputing*, 601:128206, 2024. 2

[25] Di Wu, Liu Liu, Zhou Linli, Anran Huang, Liangtu Song, Qiaojun Yu, Qi Wu, and Cewu Lu. Reartgs: Reconstructing and generating articulated objects via 3d gaussian splatting with geometric and motion constraints. In *The Thirty-ninth*

Annual Conference on Neural Information Processing Systems, 2025. 1, 2, 4, 7

[26] Di Wu, Liu Liu, Xueyu Yuan, Qiaojun Yu, Wenxiao Chen, Ruilong Yan, Yiming Tang, and Liangtu Song. Spags: Sparse-view articulated object reconstruction from single state via planar gaussian splatting. *arXiv preprint arXiv:2511.17092*, 2025. 2

[27] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20310–20320, 2024. 2

[28] 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 *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 6

[29] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *CVPR*, 2024. 8

[30] Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. In *International Conference on Learning Representations (ICLR)*, 2024. 2

[31] Mulin Yu, Tao Lu, Linning Xu, Lihan Jiang, Yuanbo Xiangli, and Bo Dai. Gsdf: 3dgs meets sdf for improved rendering and reconstruction. *arXiv preprint arXiv:2403.16964*, 2024. 2

[32] Qiaojun Yu, Junbo Wang, Wenhai Liu, Ce Hao, Liu Liu, Lin Shao, Weiming Wang, and Cewu Lu. Gamma: Generalizable articulation modeling and manipulation for articulated objects. 2024. 1

[33] Qiaojun Yu, Xibin Yuan, Yu Jiang, Junting Chen, Dongzhe Zheng, Ce Hao, Yang You, Yixing Chen, Yao Mu, Liu Liu, and Cewu Lu. Artgs:3d gaussian splatting for interactive visual-physical modeling and manipulation of articulated objects. *ArXiv*, abs/2507.02600, 2025. 1

[34] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M. Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection, 2022. 2

[35] Jiaxin Zhang, Junjun Jiang, Youyu Chen, Kui Jiang, and Xianming Liu. Cob-gs: Clear object boundaries in 3dgs segmentation based on boundary-adaptive gaussian splitting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19335–19344, 2025. 2