iTACO: Interactable Digital Twins of Articulated Objects from Casually Captured RGBD Videos

Weikun Peng[1], **Jun Lv**[2], **Cewu Lu**[2], **Manolis Savva**[1]
[1]Simon Fraser University [2]Shanghai Jiao Tong University
3dlg-hcvc.github.io/video2articulation/

Abstract

Articulated objects are prevalent in daily life. Interactable digital twins of such objects have numerous applications in embodied AI and robotics. Unfortunately, current methods to digitize articulated real-world objects require carefully captured data, preventing practical, scalable, and generalizable acquisition. We focus on motion analysis and part-level segmentation of an articulated object from a casually captured RGBD video shot with a hand-held camera. A casually captured video of an interaction with an articulated object is easy to obtain at scale using smartphones. However, this setting is challenging due to simultaneous object and camera motion and significant occlusions as the person interacts with the object. To tackle these challenges, we introduce iTACO: a coarse-to-fine framework that infers joint parameters and segments movable parts of the object from a dynamic RGBD video. To evaluate our method under this new setting, we build a dataset of 784 videos containing 284 objects across 11 categories that is 20× larger than available in prior work. We then compare our approach with existing methods that also take video as input. Our experiments show that iTACO outperforms existing articulated object digital twin methods on both synthetic and real casually captured RGBD videos.

1. Introduction

Building digital twins of articulated objects has many applications in computer animation, embodied AI, and robotics. For example, scalable pipelines for building interactable digital twins benefit robotics research, by enabling convenient training and evaluation in simulation prior to real-world deployment. However, current approaches to building interactable replicas of articulated objects impose many constraints on the input data, preventing practical and efficient acquisition of articulated objects in realistic settings. Thus, researchers still largely rely on manual annotation of articulation joint parameters [53].

Recent approaches to articulated object reconstruction and motion analysis can be categorized into three families. The first focuses on dynamic reconstruction of the shape of articulated objects [34, 44, 54, 62, 78]. Unfortunately, the final output is not interactable as it is typically a sequence of meshes instead of a kinematic structure with explicit links and joints. The second family infers joint parameters and builds part-level geometry simultaneously from two different states of the object [12, 24, 27, 29, 36, 64]. However, this requires aligning observations between states, and even small errors in alignment can lead to dramatic drops in accuracy. The third family of approaches generates articulated objects leveraging large language and multimodal models [18, 30]. At present, these approaches either rely on an external library for object retrieval or require data for fine-tuning. Retrieving objects is not easily scalable, while collecting data for fine-tuning is labor-intensive. Furthermore, generalization to real settings after fine-tuning on limited data is not easy.

By analyzing the issues of recent approaches, we summarize three desirable features for practical interactable digital twin acquisition: 1) **minimal input constraints**; 2) **explicit representation** of joint parameter and part-level geometry for easy integration into simulators; and 3) **generalization** across diverse object categories. To address these desiderata, we create digital twins of articulated objects from casually captured RGBD videos showing a human interacting with the object. For example, a human operator with a hand-held camera can first scan the object to reconstruct the object surface, and then record an interaction with the object. Such videos are easy to capture at scale and contain the necessary motion information to infer joint parameters. Compared to the problem setting in prior work, casually captured videos are far less constrained and easier to capture, the final output is easier to use for simulation, and the overall pipeline is scalable to real-world settings.

However, casually-captured video also brings more challenges. Since both the camera and the scene are dynamic, and the person occludes the object as they are manipulating it, decoupling camera motion and object part motion is challenging. Consequently, accurate estimation of joint

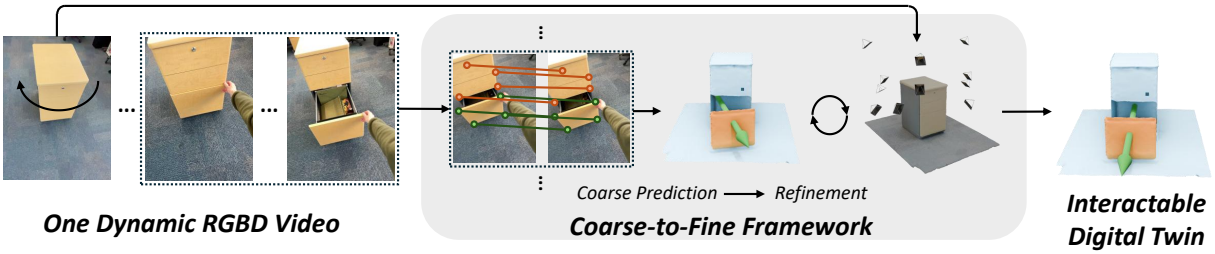

Figure 1. We propose iTACO: a coarse-to-fine framework for building interactable digital twins of articulated objects from a casually captured RGBD video. Our pipeline first predicts coarse joint parameters and movable part segmentations. The initial estimates are then refined with gradient-based optimization.

parameters and movable part segmentation is difficult. To address these challenges, we develop iTACO: a coarse-to-fine framework to build digital twins of articulated objects from dynamic RGBD videos, illustrated in Fig. 1. Our pipeline first estimates joint parameters and a movable part segmentation via image feature matching. Then, a gradient-based optimization framework refines the initial estimates against a surface point cloud representation of the object acquired through 3D reconstruction. Our pipeline explicitly parameterizes articulation and geometric parameters, satisfying the second desideratum. Moreover, to aid generalization and satisfy the third desideratum, our pipeline neither relies on an external library nor requires additional data for fine-tuning. Instead, it only uses pretrained models.

Since our problem has not been previously addressed, we build a dataset for evaluation. We generate videos for 284 synthetic objects from 11 categories in PartNet-Mobility [33] and collect real RGBD videos. Our dataset contains 20× more objects compared to prior works [24, 27], allowing a more robust evaluation of articulated object acquisition. In summary, we make the following contributions:

1. We address building interactable digital twins of articulated objects from casually captured RGBD videos, a new problem setting that is more practical than prior work.
2. We develop iTACO: a coarse-to-fine framework for acquisition of digital twins of articulated objects.
3. We collect a dataset with 20× more objects than prior datasets. Experiments show that iTACO outperforms state-of-the-art methods on both synthetic and real articulated objects across different categories.

2. Related Work

Our work is related to motion analysis and reconstruction of articulated objects, and dynamic scene understanding.

2.1. Articulated Object Reconstruction

Motion analysis and reconstruction of articulated objects is a long-standing task in computer vision and robotics. Previous motion analysis works focus on predicting joint parameters or part-level segmentation from an image or a single point cloud [11, 20, 48, 60, 63, 74, 81]. These works require training on specific datasets, which limits generalization. Previous articulated object reconstruction works focus on reconstructing the shape of articulated objects [34, 44, 62, 77, 78]. The final output of these methods is the entire shape of the object, which is **not interactable**, in particular due to the absence of a kinematic structure with joints and articulation parameters that can be driven kinematically or through physics-based simulation. In contrast, we aim to build **interactable** digital twins of articulated objects, where inferring joint parameters and part-level geometry is essential.

Another line of work attempts to infer joint parameters and build part-level geometry of articulated objects simultaneously. One problem setting uses multi-view observations of different object states [12, 24, 27, 64]. These works implicitly assume all observations have been aligned to the same coordinates. In practice, this requires either manual alignment which is not scalable or algorithmic alignment which is error-prone. Other prior work uses robots to interact with the articulated object and reconstruct articulated objects from the interaction [29, 36, 61, 75]. However, robots are expensive and may damage themselves or the environment during the interaction. Therefore, this is not a practically scalable path to build digital twins of articulated objects. Recently, researchers began exploring motion analysis from videos [14, 65, 85]. Robot-See-Robot-Do [14] recovers 3D motions of an articulated object from a monocular human demonstration video. Their problem setting is close to us, but requires the camera to be static.

Other work uses large language and multimodal models to generate a code representation or kinematic graph from a single image of an articulated object [4, 18, 25, 30]. Leveraging vision-language models, Articulate-Anything can also take image or video as input [18]. These works either rely on existing mesh libraries to retrieve object part meshes or require extra data for fine-tuning foundation models. Thus, at the current stage, this line of work still suffers from out-of-distribution issues, limiting generalization and scalability. In contrast, our method does not rely on object retrieval or fine-tuning, and is inherently more general and scalable.

In computer animation, automatic skeletal rigging has been studied for decades [1, 5, 19, 23, 45, 47, 49, 55, 70–

73, 83]. However, the skeleton model is designed to animate humans and animals, whose links are non-rigid and joints are mainly ball joints. For rigid articulated objects such as drawers and microwaves, most joints are prismatic joints and revolute joints. Thus, it is suitable to represent rigid articulated objects using skeleton models.

2.2. Dynamic Scene Reconstruction

Dynamic scene reconstruction is a challenging task in computer vision. One line of research describes a dynamic scene with a deformation field. Signed distance functions, and later neural radiance fields and 3D gaussian splatting are commonly used [2, 6, 28, 35, 38, 43, 56, 59, 66, 69, 80]. These works achieve impressive results in shape reconstruction and novel view rendering of dynamic scenes, but it is difficult to transform the deformation field to an explicit motion representation, such as a rigid transformation matrix, which is necessary for our problem.

Another line of approaches attempts to reconstruct dynamic scenes from monocular depth estimation and optical flow. Robust-CVD proposes to first estimate the depth of each frame and then optimize camera poses and depth jointly [17]. CasualSAM leverages optical flow to train a depth and movement prediction network and optimize camera parameters [84]. MegaSaM proposes an optimization framework for dynamic scene reconstruction based on DROID-SLAM [21, 52]. Recently, DUSt3R [58] provides a powerful foundation model for static scene reconstruction. There are several recent extensions to DUSt3R. MonST3R fine-tunes DUSt3R on dynamic scene datasets to recover dynamic 3D scenes [82]. CUT3R introduces a continuous 3D perception model for online dense 3D reconstruction [57]. Easi3R identifies moving objects in the video from DUSt3R's attention maps [3]. Aether proposes a unified model for reconstruction, planning, and prediction in dynamic scenes [50]. Our coarse prediction pipeline directly benefits from MonST3R [82], and our refinement framework is inspired by the optimization framework in CasualSAM [84] and MegaSaM [21].

A key sub-task in dynamic scene understanding is segmenting the 'moving map' (mask image indicating which pixels are moving in the scene at each frame in the video). Recent approaches are mainly based on optical flow or point tracking [8, 10, 13, 31, 32, 68, 76]. However, we find that point tracking methods struggle with textureless objects and are sensitive to point locations (see supplement). Therefore, we do not use these approaches.

3. Method

3.1. Problem Formulation

We define the interaction video as $\mathcal{V} = \{\mathcal{I}_t\}_{t=0}^{T}$, where T denotes the video length and each frame $\mathcal{I}_t \in \mathbb{R}^{H\times W\times 4}$ is an RGBD image with height H and width W. An object surface point cloud $\mathcal{P}^{\mathcal{O}}$ is assumed as input from a surface reconstruction stage prior to the start of interaction, as there are numerous methods for static surface reconstruction. The state of the object when reconstructing the surface is the initial state when recording the interaction video, and in practice can be captured in the same video right before the interaction begins.

We assume the video is casually captured with hand-held devices such as smartphones. Therefore, we cannot assume that the camera remains stationary during recording, nor do we assume that the camera trajectory can be easily obtained. We only assume the first video frame is aligned to the object surface coordinate system. In practice, this alignment can be achieved via point cloud registration or structure from motion. Normally, a human operator manipulates one joint at a time. Thus, we also assume only one part of the object is moving in the video. For multi-joint articulated objects, we can build each joint and part sequentially with consecutive interaction videos.

The output includes two key components: 1) joint parameters $\mathcal{J} \in \mathbb{R}^7$, including a binary value to indicate the type of joint (e.g., revolute or prismatic), as well as two triplets to indicate the position and orientation of the joint axis; and 2) movable parts of the object. Since it is challenging to segment movable parts on the 3D objects directly, we instead estimate a moving map for each video frame $\mathcal{M}_t = \{0, 1\} \in \mathbb{R}^{H\times W}$, which indicates the moving parts of the object in the video. Then, we unproject the moving map to 3D space with depth values to identify movable parts of the object. In addition, we also need to estimate both camera pose $p_t^{\text{cam}} \in SE(3)$ and the joint state $s_t \in \mathbb{R}$ at each frame.

3.2. Coarse Prediction

We first design a coarse prediction pipeline as shown in Fig. 2. The idea is to decouple the problem by firstly finding the moving part $\mathcal{M}_t$ of each frame of the video $\mathcal{V}$. Once we identify the dynamic region and static region in the video frames, we can use point motion in the static region to estimate the camera pose p_t^{cam} at each frame. We align all the observations to the same coordinate, and compute rigid transformations of movable parts of the object using points motion in the dynamic region to estimate joint parameters $\mathcal{J}$ and joint state $\{s_t\}_{t=0}^{T}$.

In practice, we use MonST3R [82] to provide a rough moving map $\mathcal{M}_t$ of each video frame. Then we unproject pixels in $\mathcal{I}_t$ to 3D point cloud $\mathcal{P}_t$ with camera intrinsics, and use LoFTr [46] to compute pixel-level feature matching between two video frames. We use feature matching in static regions identified by $\mathcal{M}_t$ to compute relative camera poses. We align all observations to the same camera coordinate system with the relative camera poses of each frame. Then,

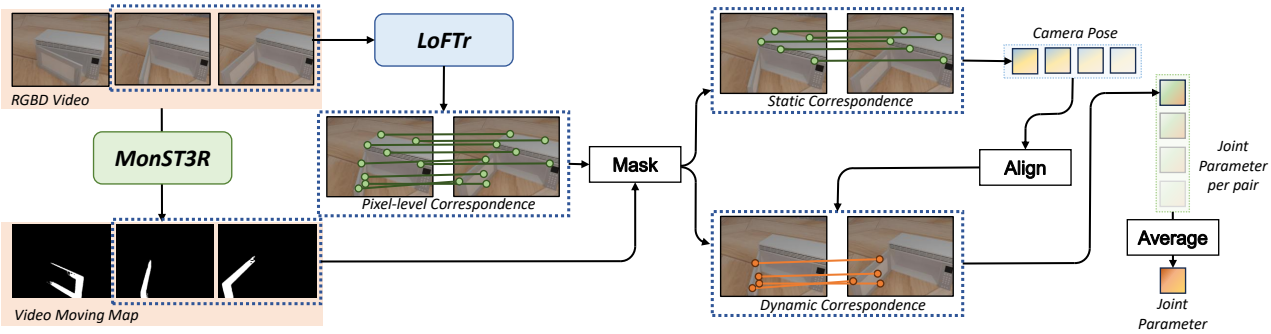

Figure 2. An overview of our coarse prediction pipeline. We first use feature matching in the static regions to estimate relative camera poses and align all observations to the same coordinate. Then, we compute the rigid transformation using feature matching in the dynamic regions to estimate joint parameters. Finally, we average out all the results to produce a joint parameter estimation.

we use feature matching in the dynamic regions to compute a rigid transformation of the movable parts of the object. We can further estimate joint parameters from this rigid transformation. We estimate joint parameters from multiple video frame pairs and average the results as the coarse prediction of the joint parameters and states.

Through the coarse prediction stage, the proposed pipeline obtains rough results for moving map segmentation and joint parameters. However, this method has two disadvantages: 1) Since the moving map is inaccurate, camera poses and joint parameters estimation are inaccurate. 2) Feature matching provides a reliable prediction only when two video frames are visually similar. In other words, the joint states do not change much, and the point motion in dynamic regions will be minimal. In this situation, it's hard to tell which joint types lead to this kind of point motion. Joint type prediction will be inaccurate as well.

3.3. Refinement

We refine the initial estimation results through gradient-based optimization. This phase is based on a simple idea: if the estimates of the joint parameters and camera poses are accurate, the object states observed in the video should transform back to the initial state. In other words, the transformed point cloud should align with the given object surface point cloud at the initial state. An overview of our pipeline is shown in Fig. 3.

Concretely, we divide the optimization target into static part and moving part. For the static part, we transform the point cloud $\mathcal{P}_t$ to the same coordinate with camera poses p_t^{cam}. We compute a **single-directional Chamfer distance** to the given object surface point cloud $\mathcal{P}^{\mathcal{O}}$ in Eq. (1).

$$\mathcal{L}_{\text{static}} = \frac{1}{T}\sum_{t=1}^{T}(1-\mathcal{M}_t)\cdot \text{Chamfer}(p_t^{\text{cam}}\mathcal{P}_t, \mathcal{P}^{\mathcal{O}}) \quad (1)$$

To compute the Chamfer distance of the moving part, we first compute the transformation matrix k_t for the moving part with $\mathcal{J}$ and s_t. Then, we transform $\mathcal{P}_t$ with camera pose p_t^{cam} and k_t. Then, we compute the **single-directional Chamfer distance** to $\mathcal{P}^{\mathcal{O}}$ in Eq. (2).

$$\mathcal{L}_{\text{dynamic}} = \frac{1}{T}\sum_{t=1}^{T}\mathcal{M}_t\cdot \text{Chamfer}(k_t p_t^{\text{cam}}\mathcal{P}_t, \mathcal{P}^{\mathcal{O}}) \quad (2)$$

Combining these two equations produces the final optimization objective $\mathcal{L} = \mathcal{L}_{\text{static}} + \mathcal{L}_{\text{dynamic}}$. Note that this computation graph is differentiable. Thus, we can use gradient-based optimization methods to optimize joint parameters, camera poses, and the moving map.

However, this naïve implementation has two problems. Firstly, it does not deal with newly observed parts of the object in the video. A newly observed part is a region that is initially hidden but becomes visible later (e.g., interior of a microwave). This region does not correspond with the object surface point cloud. Consequently, the Chamfer Distance is unreliable for point cloud alignment. Secondly, this implementation is quite inefficient. This is due to the need to optimize the probability of each pixel belonging to the moving part of each frame, which contains $T \times H \times W$ parameters in total. Considering each pixel individually is not efficient as there are millions of pixels in each frame.

To tackle the second problem our automatic part segmentation module aggregates pixels into object parts. This is more efficient and effective as the number of parts is much smaller than the number of pixels, reducing optimization complexity. Similar to Segment-Anything [16], automatic part segmentation segments the first video frame into several part regions, and then tracks these regions throughout the video using SAM2 [39]. Subsequently, instead of optimizing $\mathcal{M}_t$ on each pixel, we optimize the probability of being a moving part for each tracked segment. We define a vector $v \in (0,1)^{D\times 1}$ to represent the probability of each segment being a moving part, where D is the number of segments. To construct $\mathcal{M}_t$ for a frame, we multiply each probability value by the corresponding mask of the part. Finally, we

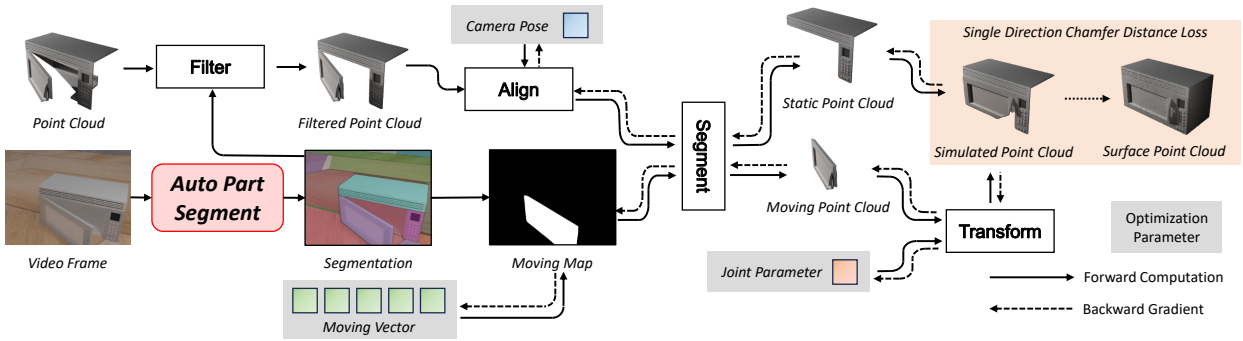

Figure 3. An overview of our refinement pipeline. We transform observations in the video back to the initial stage with camera poses and joint parameters. We then compute the chamfer distance from the transformed observation to the object surface as a loss function and optimize relevant parameters.

sum the results across different parts to construct the moving map. We leverage an existing implementation of object part segmentation for videos.

The above automatic part segmentation module is also used to identify newly observed parts, thus helping to alleviate the first problem with the naïve implementation. If a part does not appear in the first frame, then it is classified as a newly observed part. By filtering out newly observed parts in the video, we reduce the spurious correspondence issue when computing the Chamfer Distance.

Since our pipeline only estimates the moving map of the input video, we need to find the movable part of the object on $\mathcal{P}^{\mathcal{O}}$. We unproject the pixels in the moving region in the moving map of the first frame $\mathcal{M}_0$. Then, we select points on $\mathcal{P}^{\mathcal{O}}$ which are close to the unprotected moving pixels to be the movable part of the object. We use NKSR [9] to build meshes of both the static and the movable parts of the object.

4. Experiments

We aim to answer the following questions with our experiments: 1) How well can we estimate joint parameters? 2) How well can we acquire the geometry of the articulated object? 3) How do different modules and strategies of our pipeline contribute to the final results? and 4) Can our method work in real-world scenarios?

4.1. Experiment Setup

Dataset. At present, there is no benchmark for evaluating joint parameter estimation from dynamic RGBD videos. Therefore, we construct a synthetic dataset for evaluation. We select objects from 11 categories in the PartNet-Mobility dataset [33]. We simulate interaction between human hands and these objects using the SAPIEN simulator [67]. For each interaction, we generate two videos shot from two different viewpoints. Camera trajectories are generated by moving the camera towards a new target pose every 7 - 10 timesteps in the simulator. The target pose is sampled from a Gaussian distribution, where the mean value is the current camera pose and the standard deviation is set to ensure the camera can consistently look at the object. To get the object surface point cloud, we unproject the depth maps from 24 viewpoints to 3D and fuse the point cloud. For Robot-See-Robot-Do, we generate 100 views because it requires producing a NeRF and 3D Gaussian splat [14]. Please refer to the supplement for more details about the dataset construction. Since Robot-See-Robot-Do requires manually cropping and clustering 3D gaussian splats which is labor-intensive, we split the dataset into two parts. We randomly sampled 10% of test videos from our dataset to construct a smaller dataset named S-Dataset, containing 73 test videos, while L-Dataset contains the remaining videos in the original dataset.

Baselines. We select two recent works on modeling interactable digital twins of articulated objects from video input:

- **Robot-See-Robot-Do** (RSRD) [14]: a differentiable rendering pipeline that recovers 3D part motion from human demonstration videos. We follow the pipeline to first reconstruct and segment the 3D Gaussian splats of the object. Then, the input video is used to estimate the SE3 transformation of the 3D Gaussians for each part at each frame via differentiable rendering. For a fair comparison, we modify their tracking pipeline to exploit ground truth depth maps of the input video. We select the group of 3D Gaussians with the longest moving distance to be the moving part of the object. Since this pipeline does not include a geometry reconstruction module, we export 3D Gaussians and reconstruct a mesh using NKSR [9] for geometry reconstruction.
- **Articulate Anything** (AA) [18]: a system that retrieves object meshes from a library and estimates link and joint parameters from text, image, or video input. We follow the original work and use the Partnet-Mobility dataset as the object mesh library, which contains all the ground truth object meshes for our test cases, and use Gemini

Joint	Methods	Axis(rad)↓	Position(m)↓	Type(%)↓	State(rad or m)↓	Failure(%)↓
Revolute	AA [18]	0.82±0.79	0.81±0.40	40.90	N.A.	38.63
	RSRD [14]	1.16±0.52	1.18±1.21	88.88	1.05±0.59	46.66
	iTACO (Ours)	**0.32±0.56**	**0.13±0.25**	**15.90**	**0.25±0.46**	6.81
	iTACO w/o Refine	1.03±0.67	0.24±0.25	68.18	0.59±0.47	4.54
	iTACO w/o Coarse	1.26±0.29	0.34±0.26	77.27	0.69±0.45	**2.27**
	iTACO w/o Segment	0.36±0.55	0.17±0.24	18.18	0.33±0.44	6.81
Prismatic	AA [18]	0.92±0.78	N.A.	50.00	N.A.	46.66
	RSRD [14]	1.27±0.44	N.A.	40.00	0.63±0.41	36.66
	iTACO	**0.24±0.33**	N.A.	10.34	0.08±0.22	**0**
	iTACO w/o Refine	0.28±0.40	N.A.	**6.89**	0.16±0.16	**0**
	iTACO w/o Coarse	0.42±0.11	N.A.	**6.89**	0.08±0.09	**0**
	iTACO w/o Segment	0.30±0.41	N.A.	13.79	**0.07±0.07**	**0**

Table 1. Evaluation of articulation parameter estimation on S-Dataset. We report the mean and standard deviation across all the test cases within the dataset. iTACO performs best across most metrics, and ablations show the value of each component.

Methods	CD-w↓	CD-m↓	CD-s↓
AA [18]	0.11±0.22	0.59±0.73	0.07±0.18
RSRD [14]	3.39±21.50	0.60±0.60	0.17±0.44
iTACO (Ours)	**0.01±0.01**	**0.13±0.26**	**0.06±0.19**
iTACO w/o Refine	**0.01±0.01**	0.40±0.44	0.35±0.46
iTACO w/o Coarse	**0.01±0.01**	0.19±0.22	**0.06±0.16**
iTACO w/o Segment	**0.01±0.01**	0.39±0.44	0.36±0.46

Table 2. Geometry evaluation on S-Dataset. We report the mean and standard deviation across test cases within the dataset. iTACO achieves the lowest reconstruction error among all aspects.

Flash-1.5 [51] for the VLM module.

Metrics. We use a suite of metrics to evaluate both geometric reconstruction and articulation parameter estimation. Geometric accuracy is evaluated using Chamfer Distance between ground truth and estimated geometry for the whole object (CD-w), for the movable part (CD-m) and for the static part (CD-s). Articulation parameter estimation is evaluated using joint type classification error (percentage), joint axis error (radians), joint position error for revolute joints (meters), and joint state error (radians for revolute joints and meters for prismatic joints). If the method crashes or retrieves incorrect objects (for retrieval methods), we classify the output as a failure and assign $\frac{\pi}{2}$ error for joint axis and revolute joint state, and error equal to 1 for other metrics.

4.2. Articulation Parameter Estimation Results

The quantitative results on S-Dataset are in Tab. 1, and qualitative results are in Fig. 4. Results on L-Dataset are in the supplement. Our iTACO method outperforms or is comparable to state-of-the-art methods along most metrics. AA struggles with retrieving the correct object meshes. Most failure cases are due to retrieval errors, which is surprising since the retrieval library contains all the ground truth object meshes. For example, in Fig. 4 AA retrieves a lamp in the Stapler test case. In addition, the link placement and affordance detection modules appear to not be robust either, showing the challenge of our problem setting. AA selects the wheel of the table to be a movable part, and places the lid of the USB stick below the main body.

At the same time, RSRD cannot represent textureless objects accurately, likely due to limitations in using 3D Gaussian splatting. Therefore, it often fails to correctly crop and cluster 3D Gaussians into parts, even though human operators participate in this step. Most importantly, RSRD cannot handle camera motion. Thus, its movable part segmentation deviates from the ground truth significantly. In contrast, iTACO produces more accurate object articulations.

4.3. Geometry Estimation Results

We evaluate the geometry of our outputs using the Chamfer Distance to the ground truth. We sample 10000 points from the reconstructed mesh and the ground truth mesh, and compute the Chamfer Distance between them. Tab. 2 shows the results on S-Dataset. Results on L-Dataset are in the supplement. Due to retrieval errors, link placement errors, and affordance detection errors, AA has a high reconstruction error, particularly on movable parts. RSRD performs even worse, struggling with clustering 3D Gaussian splats into different parts.

4.4. Ablations

We design three ablations of our method. To demonstrate the importance of the coarse prediction module, we remove it

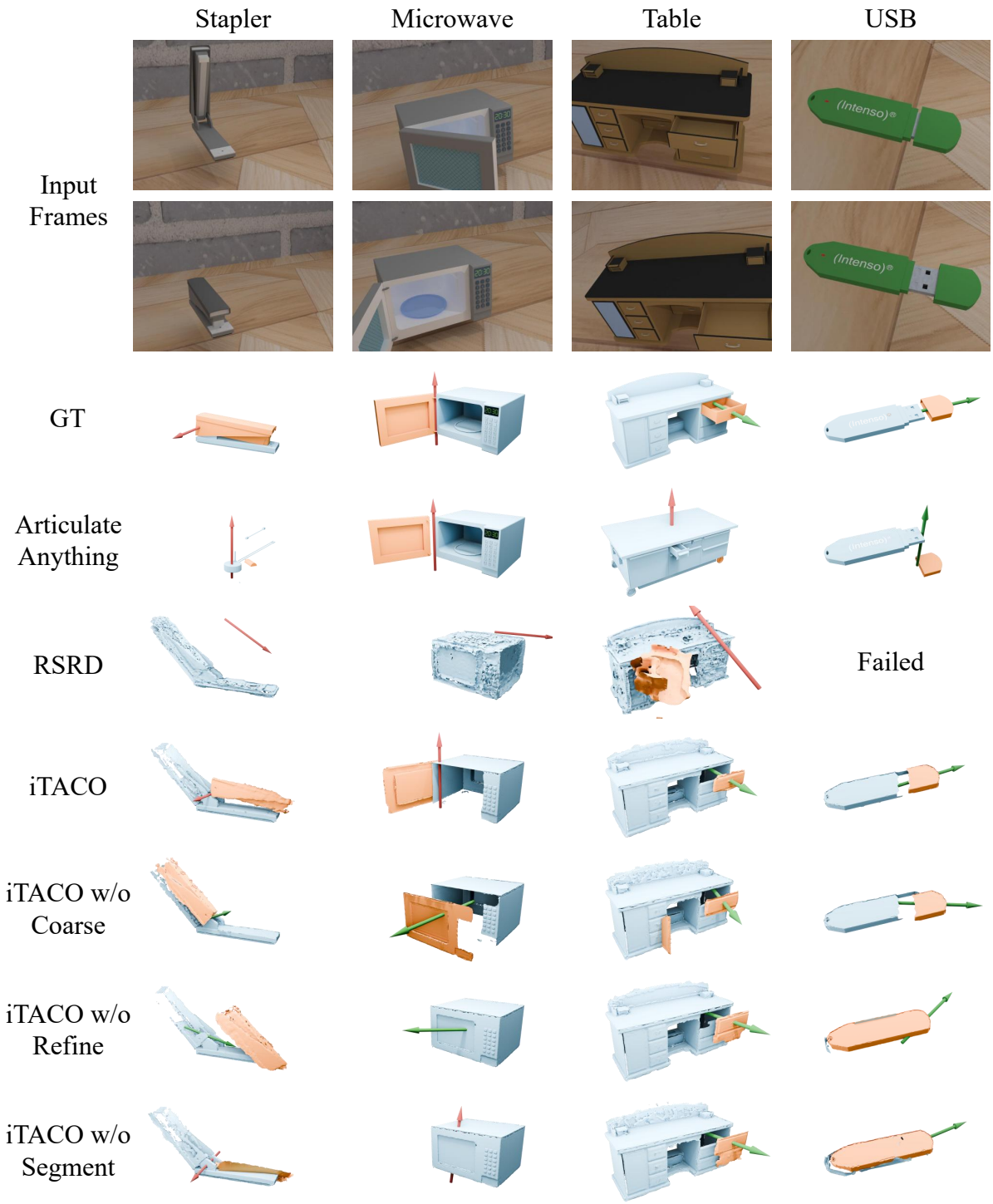

Figure 4. Qualitative results on synthetic data. The moving parts of the object are shown in orange and static parts are in blue. Prismatic joints are green and revolute joints are red. To illustrate the joint state prediction results, we render the articulated object at the end state.

from our pipeline and randomly initialize the parameters of the refinement module, named **iTACO w/o Coarse**. To illustrate the importance of the refinement module, we remove the refinement module and evaluate coarse prediction results directly, named **iTACO w/o Refine**. Lastly, to demonstrate the advantage of the automatic part segmentation strategy for the refinement process, we naïvely optimize the probability of being a moving part for each pixel at each video frame,

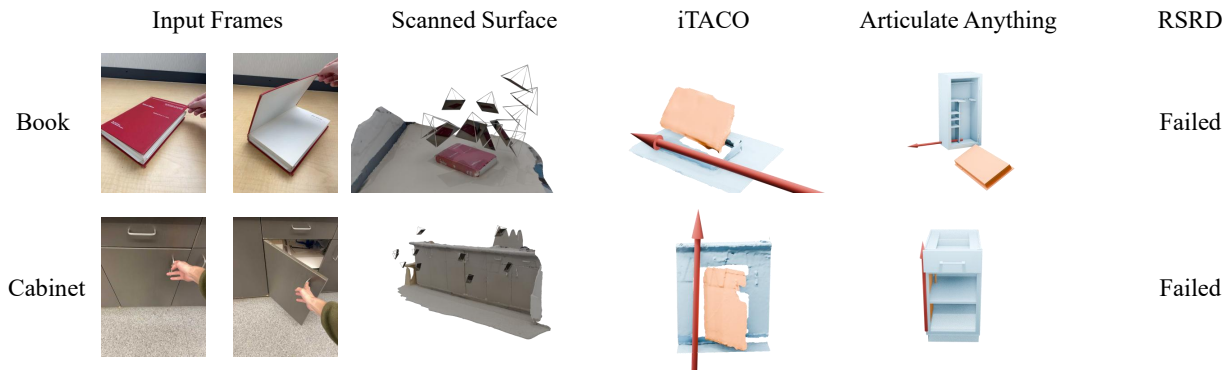

Figure 5. Qualitative results on real data. We find that Articulate Anything struggles with objects that do not exist in the mesh library, such as books. RSRD does not work well on textureless objects such as the cabinet example.

named **iTACO w/o Segment**.

In Tab. 1 we find that without refinement, coarse prediction will fail on joint type prediction for more than half of the test cases, particularly on revolute joints. As mentioned in Sec. 3, when two video frames are similar it is hard to distinguish revolute joints from prismatic joints. In Fig. 4, we also see that the coarse prediction module misclassifies the Stapler and Microwave joints. Without the refinement module it is much harder to correctly segment the movable and static parts, as shown in Tab. 2. Also, refinement cannot work effectively from a random initialization in the ablation without coarse prediction, as shown quantitatively in Tabs. 1 and 2 and qualitatively in Fig. 4.

Without automatic part segmentation, the joint prediction performance degrades as the refinement module suffers from incorrect point correspondences. For instance, the Microwave test case in Fig. 4 fails. The segmentation also deviates from the correct segmentation as shown in Tab. 2. In Fig. 4, we also see that without automatic part segmentation, our method has difficulty segmenting the moving parts and static parts accurately. This illustrates the importance of automatic part segmentation in optimizing joint parameters and the moving map from the video.

4.5. Real-World Qualitative Evaluation

We also evaluate our methods and baselines on real data. We use an iPhone 12 Pro with LiDAR camera to capture data. We use Polycam [37] to reconstruct the object surface and Record3D [40] to record the RGBD video of the human interaction with the object. To improve depth map quality, we use Prompt Depth Anything [22] to scale up the original depth map. We compute the image feature matching between the first frame of the video and all the images of the reconstruction stage with LoFTr [46]. We select the image pair with the most reliable feature matching, and compute the rigid transformation between corresponding points of the image pair to align the first video frame to the object surface coordinate system. Real videos include human hands, which interfere with MonST3R predictions for the moving map. Thus, we use Grounded SAM 2 [41, 42] to mask hands with the text prompt: "hands and arms".

From the qualitative results shown in Fig. 5, we see that AA fails on the Book test case due to the absence of similar meshes in the mesh library. In the Cabinet test case, AA retrieves reasonable meshes but predicts the wrong joint rotation direction, as the door rotates into the cabinet. RSRD fails to produce valid results due to the clustering and tracking issues on 3D gaussian. This demonstrates our method can work on real data and outperform current methods in the challenging setting of casually captured real-world RGBD video inputs.

5. Conclusion

We introduced a new problem setting for reconstructing articulated objects from casually captured RGBD videos. Our problem setting is more practical than prior works, bringing new challenges to current methods. We then developed iTACO: a coarse-to-fine framework for this problem. To evaluate our method, we collected a new dataset that contains 20× more objects than previous benchmarks. Our experiments show our method outperforms baselines, providing a practical and robust approach to acquiring digital twins of articulated objects. Nonetheless, there is still space for improvement. Better dynamic scene understanding approaches and video segmentation models can help improve our method. In addition, our pipeline does not reconstruct the interior of the object. Reconstructing the interior of the object is an interesting direction for future work.

Acknowledgements. This work was funded in part by a Canada Research Chair, NSERC Discovery Grant, and enabled by support from the Digital Research Alliance of Canada. The authors would like to thank Jiayi Liu, Xingguang Yan, Austin T. Wang, Hou In Ivan Tam, Morteza Badali for valuable discussions, and Yi Shi for proofreading.

References

[1] Ilya Baran and Jovan Popović. Automatic rigging and animation of 3D characters. *ACM Trans. Graph.*, 26(3):72–es, 2007. 2

[2] Aljaz Bozic, Michael Zollhofer, Christian Theobalt, and Matthias Nießner. DeepDeform: Learning Non-rigid RGB-D Reconstruction with Semi-supervised Data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7002–7012, 2020. 3

[3] Xingyu Chen, Yue Chen, Yuliang Xiu, Andreas Geiger, and Anpei Chen. Easi3R: Estimating Disentangled Motion from DUSt3R Without Training. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2025. 3

[4] Zoey Chen, Aaron Walsman, Marius Memmel, Kaichun Mo, Alex Fang, Karthikeya Vemuri, Alan Wu, Dieter Fox, and Abhishek Gupta. URDFormer: A Pipeline for Constructing Articulated Simulation Environments from Real-World Images. In *Robotics: Science and Systems (RSS)*, 2024. 2

[5] Yufan Deng, Yuhao Zhang, Chen Geng, Shangzhe Wu, and Jiajun Wu. Anymate: A Dataset and Baselines for Learning 3D Object Rigging. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, New York, NY, USA, 2025. Association for Computing Machinery. 2

[6] Yuanxing Duan, Fangyin Wei, Qiyu Dai, Yuhang He, Wenzheng Chen, and Baoquan Chen. 4D-Rotor Gaussian Splatting: Towards Efficient Novel-View Synthesis for Dynamic Scenes. In *Proceedings of SIGGRAPH*, 2024. 3

[7] Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981. 1

[8] Lily Goli, Sara Sabour, Mark Matthews, Brubaker Marcus, Dmitry Lagun, Alec Jacobson, David J. Fleet, Saurabh Saxena, and Andrea Tagliasacchi. RoMo: Robust Motion Segmentation Improves Structure from Motion. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2025. 3

[9] Jiahui Huang, Zan Gojcic, Matan Atzmon, Or Litany, Sanja Fidler, and Francis Williams. Neural Kernel Surface Reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4369–4379, 2023. 5, 1, 2

[10] Nan Huang, Wenzhao Zheng, Chenfeng Xu, Kurt Keutzer, Shanghang Zhang, Angjoo Kanazawa, and Qianqian Wang. Segment Any Motion in Videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 3, 2

[11] Hanxiao Jiang, Yongsen Mao, Manolis Savva, and Angel X Chang. OPD: Single-view 3D Openable Part Detection. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXIX*, pages 410–426. Springer, 2022. 2

[12] Zhenyu Jiang, Cheng-Chun Hsu, and Yuke Zhu. Ditto: Building Digital Twins of Articulated Objects from Interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1, 2

[13] Laurynas Karazija, Iro Laina, Christian Rupprecht, and Andrea Vedaldi. Learning segmentation from point trajectories. In *Advances in Neural Information Processing Systems (NIPS)*, 2024. 3

[14] Justin Kerr, Chung Min Kim, Mingxuan Wu, Brent Yi, Qianqian Wang, Ken Goldberg, and Angjoo Kanazawa. Robot See Robot Do: Imitating Articulated Object Manipulation with Monocular 4D Reconstruction. In *8th Annual Conference on Robot Learning (CoRL)*, 2024. 2, 5, 6

[15] Diederik P Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations (ICLR)*, 2015. 1

[16] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment Anything. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4015–4026, 2023. 4

[17] Johannes Kopf, Xuejian Rong, and Jia-Bin Huang. Robust Consistent Video Depth Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 3

[18] 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. 1, 2, 5, 6

[19] Peizhuo Li, Kfir Aberman, Rana Hanocka, Libin Liu, Olga Sorkine-Hornung, and Baoquan Chen. Learning skeletal articulations with neural blend shapes. *ACM Trans. Graph.*, 40(4), 2021. 2

[20] Xiaolong Li, He Wang, Li Yi, Leonidas Guibas, A Lynn Abbott, and Shuran Song. Category-Level Articulated Object Pose Estimation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[21] Zhengqi Li, Richard Tucker, Forrester Cole, Qianqian Wang, Linyi Jin, Vickie Ye, Angjoo Kanazawa, Aleksander Holynski, and Noah Snavely. MegaSaM: Accurate, Fast and Robust Structure and Motion from Casual Dynamic Videos. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 3

[22] Haotong Lin, Sida Peng, Jingxiao Chen, Songyou Peng, Jiaming Sun, Minghuan Liu, Hujun Bao, Jiashi Feng, Xiaowei Zhou, and Bingyi Kang. Prompting Depth Anything for 4K Resolution Accurate Metric Depth Estimation. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pages 17070–17080, 2025. 8

[23] Isabella Liu, Zhan Xu, Wang Yifan, Hao Tan, Zexiang Xu, Xiaolong Wang, Hao Su, and Zifan Shi. RigAnything: Template-Free Autoregressive Rigging for Diverse 3D Assets. *ACM Trans. Graph.*, 44(4), 2025. 2

[24] Jiayi Liu, Ali Mahdavi-Amiri, and Manolis Savva. PARIS: Part-level Reconstruction and Motion Analysis for Articulated Objects. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2023. 1, 2

[25] Jiayi Liu, Denys Iliash, Angel X. Chang, Manolis Savva, and Ali Mahdavi-Amiri. SINGAPO: Single Image Controlled Generation of Articulated Parts in Object. In *International Conference on Learning Representations (ICLR)*, 2025. 2

[26] Minghua Liu, Mikaela Angelina Uy, Donglai Xiang, Hao Su, Sanja Fidler, Nicholas Sharp, and Jun Gao. PartField: Learning 3D Feature Fields for Part Segmentation and Beyond. *arXiv preprint arXiv:2504.11451*, 2025. 6

[27] 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. 1, 2

[28] Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3D Gaussians: Tracking by Persistent Dynamic View Synthesis. In *International Conference on 3D Vision (3DV)*, 2024. 3

[29] Jun Lv, Qiaojun Yu, Lin Shao, Wenhai Liu, Wenqiang Xu, and Cewu Lu. SAGCI-System: Towards Sample-Efficient, Generalizable, Compositional, and Incremental Robot Learning. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 98–105. IEEE, 2022. 1, 2

[30] Zhao Mandi, Yijia Weng, Dominik Bauer, and Shuran Song. Real2code: Reconstruct articulated objects via code generation. In *International Conference on Learning Representations (ICLR)*, 2025. 1, 2

[31] Etienne Meunier and Patrick Bouthemy. Unsupervised space-time network for temporally-consistent segmentation of multiple motions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22139–22148, 2023. 3

[32] Etienne Meunier, Anaïs Badoual, and Patrick Bouthemy. EM-Driven Unsupervised Learning for Efficient Motion Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 45(4):4462–4473, 2023. 3

[33] Kaichun Mo, Shilin Zhu, Angel X. Chang, Li Yi, Subarna Tripathi, Leonidas J. Guibas, and Hao Su. PartNet: A Large-Scale Benchmark for Fine-Grained and Hierarchical Part-Level 3D Object Understanding. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 5

[34] 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 *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 12981–12991, 2021. 1, 2

[35] Richard A Newcombe, Dieter Fox, and Steven M Seitz. DynamicFusion: Reconstruction and Tracking of Non-rigid Scenes in Real-Time. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 343–352, 2015. 3

[36] Neil Nie, Samir Yitzhak Gadre, Kiana Ehsani, and Shuran Song. Structure from Action: Learning Interactions for Articulated Object 3D Structure Discovery. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1222–1229. IEEE, 2023. 1, 2

[37] Polycam. Polycam, 2025. 8

[38] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-NeRF: Neural Radiance Fields for Dynamic Scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[39] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. SAM 2: Segment Anything in Images and Videos. *arXiv preprint arXiv:2408.00714*, 2024. 4, 3

[40] Record3D. Record3d, 2025. 8

[41] Tianhe Ren and Shuo Shen. Grounded SAM 2: Ground and Track Anything in Videos, 2025. 8

[42] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing Jiang, and Lei Zhang. Grounded SAM: Assembling Open-World Models for Diverse Visual Tasks, 2024. 8

[43] Miroslava Slavcheva, Maximilian Baust, and Slobodan Ilic. SobolevFusion: 3D Reconstruction of Scenes Undergoing Free Non-rigid Motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2646–2655, 2018. 3

[44] 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 (CVPR)*, pages 5384–5395, 2024. 1, 2

[45] Chaoyue Song, Jianfeng Zhang, Xiu Li, Fan Yang, Yiwen Chen, Zhongcong Xu, Jun Hao Liew, Xiaoyang Guo, Fayao Liu, Jiashi Feng, and Guosheng Lin. MagicArticulate: Make Your 3D Models Articulation-Ready. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pages 15998–16007, 2025. 2

[46] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. LoFTR: Detector-Free Local Feature Matching with Transformers. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 3, 8, 1

[47] Mingze Sun, Junhao Chen, Junting Dong, Yurun Chen, Xinyu Jiang, Shiwei Mao, Puhua Jiang, Jingbo Wang, Bo Dai, and Ruqi Huang. DRiVE: Diffusion-based Rigging Empowers Generation of Versatile and Expressive Characters. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pages 21170–21180, 2025. 2

[48] Xiaohao Sun, Hanxiao Jiang, Manolis Savva, and Angel Xuan Chang. OPDMulti: Openable Part Detection for Multiple Objects. *arXiv preprint arXiv:2303.14087*, 2023. 2

[49] Andrea Tagliasacchi, Hao Zhang, and Daniel Cohen-Or. Curve skeleton extraction from incomplete point cloud. *ACM Trans. Graph.*, 28(3), 2009. 2

[50] Aether Team, Haoyi Zhu, Yifan Wang, Jianjun Zhou, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Chunhua Shen, Jiangmiao Pang, and Tong He. Aether: Geometric-Aware Unified World Modeling. In *Proceedings of the IEEE*

International Conference on Computer Vision (ICCV), 2025. 3

[51] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: A Family of Highly Capable Multimodal Models. *arXiv preprint arXiv:2312.11805*, 2023. 6

[52] Zachary Teed and Jia Deng. DROID-SLAM: Deep Visual SLAM for Monocular, Stereo, and RGB-D Cameras. *Advances in Neural Information Processing Systems (NIPS)*, 2021. 3

[53] Marcel Torne, Anthony Simeonov, Zechu Li, April Chan, Tao Chen, Abhishek Gupta, and Pulkit Agrawal. Reconciling Reality Through Simulation: A Real-to-Sim-to-Real Approach for Robust Manipulation. In *Robotics: Science and Systems (RSS)*, 2024. 1

[54] Wei-Cheng Tseng, Hung-Ju Liao, Lin Yen-Chen, and Min Sun. CLA-NeRF: Category-Level Articulated Neural Radiance Field. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 8454–8460. IEEE, 2022. 1

[55] Dimitrios Tzionas and Juergen Gall. Reconstructing Articulated Rigged Models from RGB-D Videos. In *European Conference on Computer Vision Workshops 2016 (ECCVW'16) - Workshop on Recovering 6D Object Pose (R6D'16)*, 2016. 2

[56] Qianqian Wang, Vickie Ye, Hang Gao, Weijia Zeng, Jake Austin, Zhengqi Li, and Angjoo Kanazawa. Shape of Motion: 4D Reconstruction from a Single Video. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2025. 3

[57] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A. Efros, and Angjoo Kanazawa. Continuous 3D Perception Model with Persistent State. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 3, 2

[58] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. DUSt3R: Geometric 3D Vision Made Easy. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3

[59] Shizun Wang, Xingyi Yang, Qiuhong Shen, Zhenxiang Jiang, and Xinchao Wang. Gflow: Recovering 4D World from Monocular Video. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 7862–7870, 2025. 3

[60] Xiaogang Wang, Bin Zhou, Yahao Shi, Xiaowu Chen, Qinping Zhao, and Kai Xu. Shape2Motion: Joint Analysis of Motion Parts and Attributes from 3D Shapes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8876–8884, 2019. 2

[61] Xi Wang, Tianxing Chen, Qiaojun Yu, Tianling Xu, Zanxin Chen, Yiting Fu, Cewu Lu, Yao Mu, and Ping Luo. Articulated Object Manipulation using Online Axis Estimation with SAM2-Based Tracking. *arXiv preprint arXiv:2409.16287*, 2024. 2

[62] Fangyin Wei, Rohan Chabra, Lingni Ma, Christoph Lassner, Michael Zollhöfer, Szymon Rusinkiewicz, Chris Sweeney, Richard Newcombe, and Mira Slavcheva. Self-supervised Neural Articulated Shape and Appearance Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15816–15826, 2022. 1, 2

[63] Yijia Weng, He Wang, Qiang Zhou, Yuzhe Qin, Yueqi Duan, Qingnan Fan, Baoquan Chen, Hao Su, and Leonidas J. Guibas. CAPTRA: CAtegory-level Pose Tracking for Rigid and Articulated Objects from Point Clouds. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 13209–13218, 2021. 2

[64] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 1, 2

[65] Abdelrhman Werby, Martin Büchner, Adrian Röfer, Chenguang Huang, Wolfram Burgard, and Abhinav Valada. Articulated Object Estimation in the Wild. In *1st Workshop on Egocentric Perception and Action for Robot Learning*, 2025. 2

[66] 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. 3

[67] 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/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 5, 2

[68] Junyu Xie, Charig Yang, Weidi Xie, and Andrew Zisserman. Moving Object Segmentation: All You Need Is SAM (and Flow). In *Proceedings of the IEEE/CVF Asian Conference on Computer Vision (ACCV)*, 2024. 3

[69] Kai Xu, Tze Ho Elden Tse, Jizong Peng, and Angela Yao. DAS3R: Dynamics-Aware Gaussian Splatting for Static Scene Reconstruction. *arXiv preprint arxiv:2412.19584*, 2024. 3

[70] Zhan Xu, Yang Zhou, Evangelos Kalogerakis, Chris Landreth, and Karan Singh. RigNet: neural rigging for articulated characters. *ACM Trans. Graph.*, 39(4), 2020. 2

[71] Zhan Xu, Yang Zhou, Li Yi, and Evangelos Kalogerakis. Morig: Motion-Aware Rigging of Character Meshes from Point Clouds. In *SIGGRAPH Asia 2022 Conference Papers*, New York, NY, USA, 2022. Association for Computing Machinery.

[72] Yajie Yan, Kyle Sykes, Erin Chambers, David Letscher, and Tao Ju. Erosion thickness on medial axes of 3D shapes. *ACM Trans. Graph.*, 35(4), 2016.

[73] Yajie Yan, David Letscher, and Tao Ju. Voxel cores: efficient, robust, and provably good approximation of 3D medial axes. *ACM Trans. Graph.*, 37(4), 2018. 3

[74] Zihao Yan, Ruizhen Hu, Xingguang Yan, Luanmin Chen, Oliver van Kaick, Hao Zhang, and Hui Huang. RPM-Net: Recurrent Prediction of Motion and Parts from Point Cloud. *ACM Transaction on Graphics (Proceedings of SIGGRAPH Asia)*, 38(6):240:1–240:15, 2019. 2

[75] Zihao Yan, Fubao Su, Mingyang Wang, Ruizhen Hu, Hao Zhang, and Hui Huang. Interaction-Driven Active 3D Reconstruction with Object Interiors. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)*, 42(6):249:1–249:12, 2023. 2

[76] Charig Yang, Hala Lamdouar, Erika Lu, Andrew Zisserman, and Weidi Xie. Self-supervised Video Object Segmentation by Motion Grouping. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021. 3

[77] Gengshan Yang, Deqing Sun, Varun Jampani, Daniel Vlasic, Forrester Cole, Huiwen Chang, Deva Ramanan, William T Freeman, and Ce Liu. LASR: Learning Articulated Shape Reconstruction from a Monocular Video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[78] Gengshan Yang, Minh Vo, Natalia Neverova, Deva Ramanan, Andrea Vedaldi, and Hanbyul Joo. BANMo: Building Animatable 3D Neural Models from Many Casual Videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1, 2

[79] Yunhan Yang, Yukun Huang, Yuan-Chen Guo, Liangjun Lu, Xiaoyang Wu, Edmund Y Lam, Yan-Pei Cao, and Xihui Liu. SAMPart3D: Segment Any Part in 3D Objects. *arXiv preprint arXiv:2411.07184*, 2024. 6

[80] 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. 3

[81] 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. In *2024 International Conference on Robotics and Automation (ICRA)*. IEEE, 2024. 2

[82] Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. MonST3R: A Simple Approach for Estimating Geometry in the Presence of Motion. *International Conference on Learning Representations (ICLR)*, 2025. 3, 1, 2

[83] Jia-Peng Zhang, Cheng-Feng Pu, Meng-Hao Guo, Yan-Pei Cao, and Shi-Min Hu. One Model to Rig Them All: Diverse Skeleton Rigging with UniRig. *ACM Trans. Graph.*, 44(4), 2025. 3

[84] Zhoutong Zhang, Forrester Cole, Zhengqi Li, Michael Rubinstein, Noah Snavely, and William T. Freeman. Structure and Motion from Casual Videos. In *Computer Vision – ECCV 2022*, pages 20–37, Cham, 2022. Springer Nature Switzerland. 3

[85] Hongyi Zhou, Xiaogang Wang, Yulan Guo, and Kai Xu. MonoMobility: Zero-Shot 3D Mobility Analysis from Monocular Videos. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2025. 2