Articulation in Prime: Primitive-Based Articulated Object Understanding from a Single Casual Video

Arslan Artykov, Tom Ravaud, Nicolás Violante, Vincent Lepetit

LIGM, CNRS, Univ Gustave Eiffel, ENPC, Institut Polytechnique de Paris, France

Abstract

Retrieving the 3D kinematics of articulated objects from monocular video is a fundamental challenge in computer vision. Existing methods rely on complex video setups or cues such as long-term point tracking or wide-baseline matching, but are frequently brittle under severe occlusions, rapid camera ego-motion, or weak local features. Learning-based methods, meanwhile, struggle to generalize beyond their training categories. We propose a category-agnostic optimization framework that treats articulated object understanding as a primitive-fitting problem. Geometric primitives serve as a proxy representation that avoids the pitfalls of unstable point tracks; a novel mechanism organizes them into coherent parts constrained by revolute and prismatic joints. Our formulation jointly optimizes part segmentation and joint parameters, recovering kinematics from a single casually captured video. A visibility-aware procedure handles partial observations and occlusions inherent to real-world data. We also propose the AiP-synth and AiP-real benchmarks, featuring significant camera motion and heavy occlusions, and outperform existing methods. Project page: `https://aartykov.github.io/Articulation-in-Prime/`

1 Introduction

Articulated objects are composed of rigid parts connected by joints, which constrain the parts to move according to the type of joint, e.g., prismatic and revolute. Understanding articulated objects from images is a fundamental problem in computer vision, where the goal is to segment the object into articulated parts and recover joint parameters.

Early approaches required elaborate acquisition protocols impractical for real-world use, such as multi-view image sets at two distinct articulation states (19; 42), or a 3D object scan paired with an interaction video (13; 1). We instead focus on monocular video of an object exhibiting articulated motion, which simplifies acquisition but introduces technical challenges such as camera ego-motion and object occlusions. To address these, existing approaches (20; 2) rely on long-term point tracks or wide-baseline matches to estimate part motion, leading to brittle reconstructions when local features fail. Learning-based alternatives struggle to generalize to unseen object categories, even when trained on synthetic data with ground-truth annotations.

We formulate articulated object understanding as a primitive-fitting problem over an input video. Our approach leverages the global structure of primitives as a *proxy representation* to guide optimization, making it more robust than tracking-based methods and generalizing across a diverse set of objects. For each frame, we backproject depth into 3D points, then optimize primitives to fit the resulting point clouds and move according to predicted joints. A novel mechanism organizes primitives into parts constrained by articulated motions, and a visibility-aware procedure handles partial observations inherent to real-world captures. We evaluate on existing benchmarks and on AiP-synth and AiP-real, two new benchmarks we introduce featuring large camera motions and heavy occlusions.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

2 Related Work

Learning-based articulated object understanding. Early methods (26) extend implicit representations (29) to category-level modeling by embedding joint angles in a learned manifold. CAPTRA (41) tracks articulated motion directly from point cloud sequences, while ANCSH (18) generalizes NOCS (40) to represent canonical configurations of articulated objects. However, unlike our approach, these methods are typically category-specific. Ota et al. (28) identify movable components and validates them through physical interaction. Ditto (11) estimates geometry and relative state changes from multi-view 3D point clouds.

Articulated object reconstruction from a point cloud video. Optimization-based methods on point cloud sequences such as ReArt (20) and Chat et al. (4) (code not available) recover a hierarchy of piecewise rigid bodies that generalizes across categories. Other methods (36) rely on complete point clouds of the object at each timestep. To improve real-world applicability, we instead limit our input to single-view point clouds. P^3-Net (35) uses recurrent neural networks for part and motion estimation; other unsupervised approaches (48) focus only on part segmentation.

Articulated object reconstruction from multi-view input. Watch-It-Move (27) uses geometric cues across multiple video sequences to infer object articulation. To mitigate the requirement for dense temporal data, PARIS (19) and DTA (42) use only two sets of images of fixed object states. More recently, Wu et al. (44) leverage 3DGS (12) for high-fidelity geometry reconstruction; however, their method does not address part and joint estimation. Kim et al. (14) introduces an end-to-end differentiable optimization framework that recovers part-level 3D geometry, screw-based joint axes, and joint states directly from multi-view RGB images captured at different articulation states. Other works (32; 1; 13) estimate joint parameters by combining object scans with interaction videos. But in contrast to our approach, this requires a separate acquisition stage to obtain a scan.

Articulated object understanding from a single video input. Similarly to our method, REACTO (37), Artykov et al. (2), Articulate-Anything (17), Artipoint (43), and FreeArtGS (5) (code not available) operate on single-video inputs. REACTO reconstructs surface geometry without estimating joint parameters, while Articulate-Anything estimates joint parameters for single-part objects but bypasses geometry reconstruction by retrieving meshes from an external database. Artipoint leverages hand-centric priors and persistent 2D trajectories for keypoint tracking for joint parameter extraction.

Object decomposition into primitives. Representing an object as a set of geometric primitives (38; 46; 6) is a classical problem in computer vision and graphics. Optimization-based methods (21; 22; 8), optimize the parameters of the primitives directly, while learning-based methods (38; 31; 6; 46; 7) predict them from an observation of the object. We choose the popular superquadrics (7; 21; 31), as they offer a good trade-off between expressiveness and compactness of representation. Unlike these methods, which target accurate object reconstruction, our approach uses primitives as a *proxy* for structures with geometric and dynamic coherence.

3 Method

The input to our method is a sequence of depth maps $\mathbf{D}_{t\leq T}$ of an object undergoing articulated motion and corresponding camera poses extracted from a video of T frames. For real sequences, we use ViPE (10) to obtain depth maps and camera poses and like previous works (2; 1; 32), we use SAM2 (34) to get the object masks.

We backproject the sequence of depth maps into a sequence of point clouds $\mathbf{X}_t = \{\mathbf{x}_{t,n}\}_{n\leq N}$, where the 3D points $\mathbf{x}_{t,n}$ are expressed in world coordinates. Point cloud $\mathbf{X}_t$ contains only the points of the object seen in frame t. Thus, each $\mathbf{X}_t$ contains only a portion of the surface of the object due to camera movement and occlusions.

Parts and Primitives. We use convex superquadrics for the primitives. A superquadric is parameterized by a 3D rotation $\mathbf{R}_q$ and translation $\mathbf{t}_q$, its scale $\mathbf{s}_q \in \mathbb{R}^3$, and $\varepsilon_q \in [0.1, 1.9]^2$, the 2-vector controlling its shape (more details are given in the appendix (Sec. A)). These primitives are assigned to parts via a soft differentiable assignment, and the motion of each primitive is a weighted aggregation of the part motions. At convergence, a primitive is assigned to only one part. We consider a set $\mathcal{P} = \{\mathbf{P}_p\}_{p\leq P}$ of P parts and a set $\mathcal{Q} = \{\mathbf{Q}_q\}_{q\leq Q}$ of Q primitives. During optimization, the

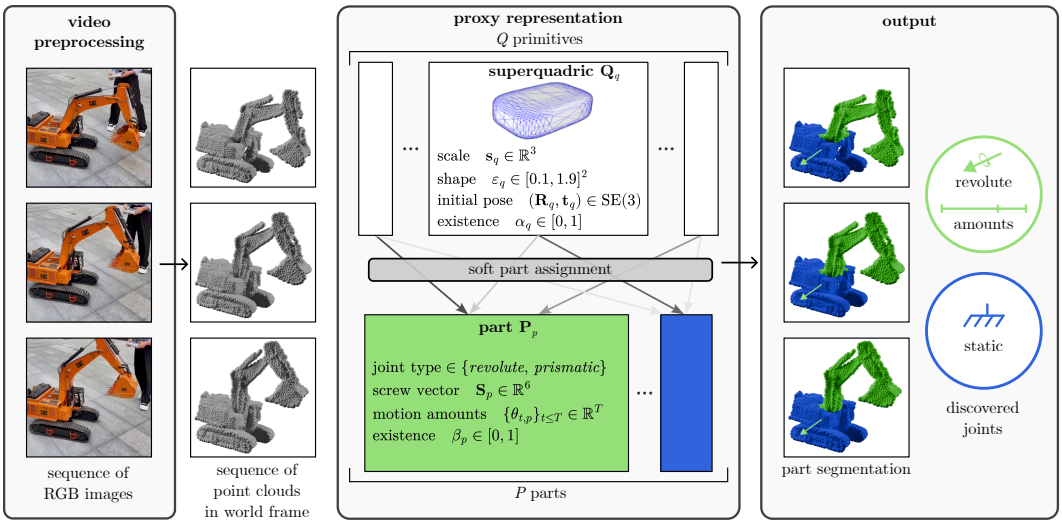

Figure 1: **Overview.** Given a sequence of depth maps and camera poses extracted from a video, we backproject the frames into a sequence of partial point clouds in world coordinates. We jointly optimize a set of superquadric primitives and part assignments, where each primitive is softly assigned to a part via a differentiable allocation, and each part carries its own joint parameters (revolute or prismatic) and per-timestep motion amounts. The optimization is driven by reconstruction and scene flow losses, with regularization encouraging a sparse part and primitive decomposition.

probability $v_{q,p}$ for primitive $\mathbf{Q}_q$ to belong to part $\mathbf{P}_p$ is computed as:

$$v_{q,p} = \frac{\exp(h_{q,p})}{\sum_{p'} \exp(h_{q,p'})} , \tag{1}$$

where $h_{q,p}$ are the entries of an assignment matrix. Because the numbers of parts and primitives are typically small, we directly optimize on the $h_{q,p}$ values.

To segment the point clouds into parts, we first introduce the probability $w_{t,n,q}$ for point $\mathbf{x}_{t,n}$ to belong to primitive $\mathbf{Q}_q$, which we compute as:

$$w_{t,n,q} = \frac{\exp(\mathbf{f}_{t,n}.\mathbf{g}_q)}{\sum_{q'} \exp(\mathbf{f}_{t,n}.\mathbf{g}_{q'})} , \tag{2}$$

where $\mathbf{g}_q$ is an optimized feature vector for primitive $\mathbf{Q}_q$ and $\mathbf{f}_{t,n}$ is a feature vector for point $\mathbf{x}_{t,n}$ predicted by a small MLP which we optimize together with the other parameters. This MLP takes the point's 3D coordinates as input. Since the number of points is very large, it is more efficient to optimize on the MLP parameters and the feature vectors of the primitives instead of directly optimizing an assignment matrix.

Joint Parameters. Each part $\mathbf{P}_p$ has joint parameters. To half of the parts, we assign a prismatic joint, and a revolute joint to the remaining parts. We optimize on more parts than we expect to find on the target object. The superfluous parts will not be assigned any primitive after optimization and can thus be discarded trivially, allowing us to automatically identify the number of parts and the nature of their joint.

We parameterize the joint parameters using twist vectors from screw theory. A twist vector $\mathbf{S}_{\text{rev}}$ for a revolute motion can be written as $\mathbf{S}_{\text{rev}} = [\omega, (-\omega \times \mathbf{q}_\perp)]$ with $\mathbf{q}_\perp = \mathbf{q} - (\mathbf{q}.\omega)\omega$, under the constraint that $\|\omega\| = 1$, so we optimize over 3-vectors ω and $\mathbf{q}$. A twist vector $\mathbf{S}_{\text{prism}}$ for a prismatic motion can be written as $\mathbf{S}_{\text{prism}} = [0, \mathbf{v}]$ with $\|\mathbf{v}\| = 1$, and we parameterize it with $\mathbf{v}$. Each part has thus a screw vector $\mathbf{S}_p$, and a motion amount per timestep $\theta_{t,p}$. We can compute the transformation matrix $\mathbf{T}_{t,p}$ (3D rotation and translation) for part p at time t by applying Rodrigues' formula to $(\theta_{t,p}\mathbf{S}_p)$. If at the end of the optimization the motion amounts for a part are small enough, then the part is considered static. We can now express how a primitive q moves by computing its screw vector as:

$$\mathbf{T}_{t,q} = \left(\sum_p v_{q,p} \theta_{t,p} \mathbf{S}_p \right) .[\mathbf{R}_q, \mathbf{t}_q] , \tag{3}$$

by taking the linear combination of the parts' screw vector and applying it to the rotation and translation of the primitive.

Objective Function. We optimize all parameters using Adam (15) over a global objective function that balances observation and regularization terms:

$$\mathcal{L} = \mathcal{L}_{\text{obs}} + \mathcal{L}_{\text{reg}} \,. \tag{4}$$

Optimization details and weights for each loss term are given in the appendix (Sec. B.2); we do not show the weights in the equations below for simplicity.

The observation loss $\mathcal{L}_{\text{obs}}$ measures how well the primitives fit the 3D points and vice-versa, and how well the primitive motions fit the scene flow between consecutive time steps:

$$\mathcal{L}_{\text{obs}} = \mathcal{L}_{\text{rec}}^{\mathcal{Q}\to\mathcal{X}} + \mathcal{L}_{\text{rec}}^{\mathcal{X}\to\mathcal{Q}} + \mathcal{L}_{\text{flow}} \,. \tag{5}$$

Reconstruction Loss. Each superquadric is represented by points $\mathbf{y}$ sampled from its surface (please refer to the appendix (Sec. A.2) for sampling details). The term $\mathcal{L}_{\text{rec}}^{\mathcal{Q}\to\mathcal{X}}$ ensures that the primitives remain close to observed points. In contrast with previous work using superquadrics (31; 7), we use a single camera and the scene is dynamic, so we do not observe complete point clouds, only points that are visible from the camera. Simply taking $\mathcal{L}_{\text{rec}}^{\mathcal{Q}\to\mathcal{X}}$ as the Chamfer distance between the primitives and the point clouds would result in a biased estimate as many points on the superquadrics are not explained by these partial point clouds. To solve this issue, we consider only the points on the superquadrics that are visible from the camera

$$\mathcal{L}_{\text{rec}}^{\mathcal{Q}\to\mathcal{X}} = \frac{1}{\sum_{t\le T}\sum_{q\le Q}\sum_{\mathbf{y}\in\mathcal{V}_t(\mathbf{T}_{t,q}.\mathbf{Q}_q)}\gamma_{t,\mathbf{y},q}} \sum_{t\le T}\sum_{q\le Q}\sum_{\mathbf{y}\in\mathcal{V}_t(\mathbf{T}_{t,q}.\mathbf{Q}_q)} \gamma_{t,\mathbf{y},q} d(\mathbf{y}, \mathbf{X}_t) \,, \tag{6}$$

where

$$d(\mathbf{y}, \mathbf{X}) = \min_{\mathbf{x}\in\mathbf{X}} \|\mathbf{y} - \mathbf{x}\|_2^2 \,, \tag{7}$$

and $\mathcal{V}_t(\mathbf{T}.\mathbf{Q})$ is the visible part of primitive $\mathbf{Q}$ under pose $\mathbf{T}$ at time t. $\gamma_{t,\mathbf{y},q} \in [0,1]$ is the probability that point $\mathbf{y}$ on primitive $\mathbf{Q}_q$ is not occluded by any other primitive. Inspired by NeRF/3DGS (12; 23), we compute it as

$$\gamma_{t,\mathbf{y},q} = \alpha_q \prod_{q'\in\mathcal{O}_t(\mathbf{y})} (1 - \alpha_{q'}) \,, \tag{8}$$

where $\mathcal{O}_t(\mathbf{y})$ is the set of primitives that occlude point $\mathbf{y}$ at time t and α_q is the probability of existence of primitive $\mathbf{Q}_q$. We optimize it directly using $\mathcal{L}_{\text{prim-existence}}$ introduced below (Eq.(17)). In practice, we obtain $\mathcal{O}_t(\mathbf{y})$ by rendering the primitives into a Z-buffer (see the appendix (Sec. A.3) for more details).

Conversely, the term $\mathcal{L}_{\text{rec}}^{\mathcal{X}\to\mathcal{Q}}$ ensures that observed points are explained by primitives:

$$\mathcal{L}_{\text{rec}}^{\mathcal{X}\to\mathcal{Q}} = \frac{1}{T.N} \sum_{t\le T}\sum_{n\le N}\sum_{q\le Q} w_{t,n,q} d_t(\mathbf{x}_{t,n}, \mathbf{T}_{t,q}.\mathbf{Q}_q) \,, \tag{9}$$

where $d_t(\mathbf{x}, \mathbf{T}.\mathbf{Q})$ is the distance between 3D point $\mathbf{x}$ and primitive $\mathbf{Q}$ under pose $\mathbf{T}$ at time t:

$$d_t(\mathbf{x}, \mathbf{T}.\mathbf{Q}) = \min_{\mathbf{y}\in\mathcal{V}_t(\mathbf{T}.\mathbf{Q})} \|\mathbf{x} - \mathbf{y}\|_2^2 \,. \tag{10}$$

Flow Loss. The term $\mathcal{L}_{\text{flow}}$ ensures that the parts move according to the scene flow. Our ablation study shows it is crucial to get good results. For each 3D point $\mathbf{x}_{t,n}$, we predict its location $\hat{\mathbf{x}}_{t,n}$ for the next time step from the part poses $\mathbf{T}_{p,t}$ over time:

$$\hat{\mathbf{x}}_{t+1,n,p} = \mathbf{T}_{p,t+1}\mathbf{T}_{p,t}^{-1}\mathbf{x}_{t,n} \,, \tag{11}$$

$\mathcal{L}_{\text{flow}}$ can then be computed as:

$$\mathcal{L}_{\text{flow}} = \frac{1}{(T-1).N} \sum_{t<T}\sum_{n\le N}\sum_{p\le P} u_{t,n,p} \|(\hat{\mathbf{x}}_{t+1,n,p} - \mathbf{x}_{t,n}) - \mathbf{f}_{t,n}\|_1 \,, \tag{12}$$

where $u_{t,n,p} = \sum_{q \leq Q} v_{q,p} w_{t,n,q}$ is the probability for point $\mathbf{x}_{t,n}$ to belong to part p and $\mathbf{f}_{t,n}$ is the scene flow at point $\mathbf{x}_{t,n}$. In practice, we use GMSF (47) to compute the scene flow from the sequence of point clouds.

Regularization Loss. Regularization loss $\mathcal{L}_{\text{reg}}$ mostly aims at minimizing the number of parts, the number of primitives per part, and at encouraging smooth articulated motions:

$$\mathcal{L}_{\text{reg}} = \mathcal{L}_{\text{parts}} + \mathcal{L}_{\text{prims}} + \mathcal{L}_{\text{motion}} + \mathcal{L}_{\text{match}} \,. \tag{13}$$

Term $\mathcal{L}_{\text{parts}} = \mathcal{L}_{\text{part-sparsity}} + \mathcal{L}_{\text{part-existence}}$:

$$\mathcal{L}_{\text{part-sparsity}} = \left(\frac{1}{P} \sum_{p \leq P} \sqrt{\frac{1}{Q} \sum_{q \leq Q} v_{q,p}} \right)^2 \,, \tag{14}$$

and, like previous works did to encourage primitives (46; 7) to "exist" or not, we use a binary cross-entropy loss:

$$\mathcal{L}_{\text{part-existence}} = -\frac{1}{P} \sum_{p \leq P} \left[\hat{\beta}_p \log \beta_p + (1 - \hat{\beta}_p) \log (1 - \beta_p) \right] . \tag{15}$$

where we optimize directly on the β_p, the probabilities for parts of existing and $\hat{\beta}_p = 1_{\frac{1}{Q} \sum_{q \leq Q} v_{q,p} > \tau_\beta}$. At the end of optimization, we can check if a part p actually exists by similarly thresholding β_p.

Term $\mathcal{L}_{\text{prims}} = \mathcal{L}_{\text{prim-sparsity}} + \mathcal{L}_{\text{prim-existence}}$ plays a similar role for primitives:

$$\mathcal{L}_{\text{prim-sparsity}} = \left(\frac{1}{Q} \sum_{q \leq Q} \sqrt{\frac{1}{T.N} \sum_{t \leq T} \sum_{n \leq N} w_{t,n,q}} \right)^2 \tag{16}$$

limits the number of primitives per part, and

$$\mathcal{L}_{\text{prim-existence}} = -\frac{1}{Q} \sum_{q \leq Q} \left[\hat{\alpha}_q \log \alpha_q + (1 - \hat{\alpha}_q) \log (1 - \alpha_q) \right] \tag{17}$$

encourages the primitives to "exist" or not. We optimize on the α_q, the probabilities for primitives of existing, with $\hat{\alpha}_q = 1_{\sum_{t \leq T} \sum_{n \leq N} w_{t,n,q} > \tau_\alpha}$. At the end of optimization, we can check if a primitive q actually exists by similarly thresholding α_q by τ_α.

Term $\mathcal{L}_{\text{motion}}$ limits the magnitudes of the linear or angular velocities of the parts:

$$\mathcal{L}_{\text{motion}} = \sum_{t < T} \sum_{p \leq P} (\theta_{t+1,p} - \theta_{t,p})^2 \,. \tag{18}$$

Finally, **term** $\mathcal{L}_{\text{match}}$ encourages primitives to be matched to only one part since minimizing the entropy of the $v_{q,p}$'s encourages them to tend to a one-hot vector:

$$\mathcal{L}_{\text{match}} = -\frac{1}{Q} \sum_{q \leq Q} \sum_{p \leq P} v_{q,p} \log v_{q,p} \,. \tag{19}$$

4 Experiments

4.1 Baseline Methods

Existing approaches in articulated object modeling typically rely on multi-view setups or 3D scans paired with aligned interaction sequences, which complicates the identification of directly comparable baselines. To evaluate our approach, we compare it against the following methods.

Video2Articulation (32) is an optimization-based approach that inputs a 3D object scan along with a scan-aligned interaction video and segments the object's dynamic part and estimates joint parameters.

Articulate-Anything (17) inputs a single video, but it assumes access to a mesh database from which object parts are retrieved, and it does not predict part labels or per-timestep motion values, so we denote these entries as "N/A" in our tables.

Table 1: **Quantitative results on the AiP-synth dataset.** 'N/A' means the method does not predict the metric. 'F' means the method fails on the category. Best results are in **bold**.

Joint Type	Method	mIoU ↑	Axis ↓ (°)	Position ↓ (cm)	State ↓ (° / cm)	Type Acc. ↑ (%)
Revolute	Articulate-Anything (17)	N/A	69.82 ± 37.19	70.66 ± 53.23	N/A	33.3
	Video2Articulation (32)	0.48 ± 0.31	51.20 ± 34.99	80.95 ± 34.20	57.77 ± 30.02	25.0
	ReArt (20)	0.17 ± 0.31	74.07 ± 30.22	80.71 ± 35.22	80.46 ± 24.18	25.0
	Ours	$\mathbf{0.85 \pm 0.20}$	$\mathbf{0.52 \pm 1.38}$	$\mathbf{0.58 \pm 1.45}$	$\mathbf{1.45 \pm 2.86}$	**100.0**
Prismatic	Articulate-Anything (17)	N/A	45.58 ± 44.42	–	N/A	50.0
	Video2Articulation (32)	0.59 ± 0.21	51.46 ± 33.06	–	18.63 ± 7.24	75.0
	ReArt (20)	F	F	–	F	F
	Ours	$\mathbf{0.94 \pm 0.04}$	$\mathbf{(4.5 \pm 0.2) \times 10^{-3}}$	–	$\mathbf{0.14 \pm 0.23}$	**100.0**

ReArt (20) is the most comparable to our approach, as it takes a dynamic point cloud sequence as input. It segments object parts in a canonical frame and propagates them to other timesteps using optimized joint parameters.

Artipoint (43) leverages human-object interaction to identify affordance points and uses long-term point tracking to follow moving parts. Similar to Articulate-Anything, it does not estimate part labels or motion states. Since Artipoint requires human-object interaction, we restrict its evaluation to real-world sequences.

4.2 Datasets

Video2Articulation-S. We evaluate on the simulated dataset introduced by Video2Articulation (32), which consists of 73 videos spanning 11 categories from the PartNet-Mobility dataset (24).

Arti4D. We evaluate on the Arti4D dataset introduced by Artipoint (43), which consists of object interaction sequences captured in real-world settings. The dataset comprises four scenes characterized by repetitive object instances. Consequently, we selected 11 representative objects from across the full set of scenes for our evaluation.

AiP-synth. We introduce a novel dataset comprising 16 challenging categories. We curated a diverse selection of objects—including bottles, cameras, and globes—specifically to evaluate the robustness of our framework across a wide spectrum of kinematic and articulation configurations. We utilize the SAPIEN (45) environment to render RGB-D sequences of objects sourced from the PartNet-Mobility (24) dataset. To ensure diversity in perspective, our sequences incorporate two distinct camera trajectories: ping-pong and circular arc. The figures of the representative objects from the dataset and representative figures of camera trajectories are provided in the appendix (Sec. C.1).

AiP-real. To assess performance in unconstrained real-world environments, we curated a real-world dataset. To demonstrate the robustness and accuracy of our framework, we supplemented our own captures with diverse sequences sourced from YouTube, including challenging objects such as globes and excavators.

4.3 Metrics

We evaluate object part segmentation using the mean Intersection-over-Union (mIoU) metric. For joint axis estimation, we report the sign-agnostic axis–angle error (in degrees), and for pivot localization, we use the point-to-line distance (in centimeters). We further evaluate the motion state using the mean absolute error (L1) between the predicted and ground-truth motion magnitudes, reported in degrees or centimeters as appropriate. Finally, we report joint type accuracy as the percentage of correctly predicted joint types.

4.4 Results

As illustrated in Tables 1 and 2 and Figures 3 and 4, our method significantly outperforms competing baselines on both the AiP-synth and AiP-real datasets. Our framework achieves substantial performance improvements despite utilizing only a single casual video as input. On synthetic sequences, which provide perfect depth maps, scene flow, and camera poses, the proposed method actually attains perfect accuracy across several evaluation metrics. In contrast, existing baseline methods

Table 2: **Quantitative results on the AiP-real dataset.** 'N/A' means the method does not predict the metric. 'F' means the method fails on the category. Best results in **bold**.

Metrics	Method	Box	Chair	Sliding Box	Globe	Excavator	Mean
mIoU ↑	Articulate-Anything	N/A	N/A	N/A	N/A	N/A	N/A
	Video2Articulation	0.49	0.02	**0.85**	0.00	0.55	0.38
	ReArt	F	0.35	F	F	F	0.35
	Artipoint (43)	N/A	N/A	N/A	N/A	N/A	N/A
	Ours	**0.91**	**0.91**	0.71	**0.86**	**0.89**	**0.86**
Axis (°) ↓	Articulate-Anything	90.00	F	90.00	F	F	90.00
	Video2Articulation	89.02	10.68	81.84	15.21	**10.17**	41.38
	ReArt	F	81.25	F	F	F	81.25
	Artipoint (43)	**7.54**	86.09	35.90	F	34.38	40.98
	Ours	10.18	**3.98**	**6.89**	**1.13**	10.49	**6.53**
Position (cm) ↓	Articulate-Anything	28.45	F	–	F	F	28.45
	Video2Articulation	135.69	30.16	–	5.20	5.09	44.04
	ReArt	F	100.0	–	F	F	100.0
	Artipoint (43)	4.07	63.86	–	F	6.94	24.96
	Ours	**1.35**	**6.47**	–	**3.01**	**2.23**	**3.27**
Type Accuracy (%) ↑	Articulate-Anything	**100.0**	F	0.00	F	F	50.0
	Video2Articulation	**100.0**	**100.0**	0.00	**100.0**	**100.0**	80.0
	ReArt	F	0.00	F	F	F	0.00
	Artipoint (43)	**100.0**	**100.0**	0.00	F	**100.0**	75.0
	Ours	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**

Table 3: **Quantitative results on the Video2Articulation-S dataset.** 'N/A' means the method does not predict the metric. 'F' means the method fails on the category. Best results are in **bold**.

Joint Type	Method	mIoU ↑	Axis ↓ (°)	Position ↓ (cm)	State ↓ (° / cm)	Type Acc. ↑ (%)
Revolute	Articulate-Anything (17)	N/A	51.43 ± 44.54	33.13 ± 32.01	N/A	85.7
	Video2Articulation (32)	0.73 ± 0.22	9.72 ± 25.13	8.41 ± 5.84	8.65 ± 15.32	88.9
	ReArt (20)	0.36 ± 0.36	54.58 ± 34.84	66.06 ± 35.20	75.71 ± 21.92	55.5
	Ours	$\mathbf{0.93 \pm 0.05}$	$\mathbf{(2.8 \pm 6.5) \times 10^{-3}}$	$\mathbf{(1.5 \pm 1.1) \times 10^{-3}}$	$\mathbf{0.14 \pm 0.27}$	**100.0**
Prismatic	Articulate-Anything (17)	N/A	90.00 ± 0.00	–	N/A	0
	Video2Articulation (32)	0.58 ± 0.34	6.83 ± 0.40	–	4.06 ± 0.44	**100.0**
	ReArt (20)	F	F	–	F	F
	Ours	$\mathbf{0.88 \pm 0.02}$	$\mathbf{(3.1 \pm 6.3) \times 10^{-3}}$	–	$\mathbf{3.00 \pm 0.50}$	**100.0**

exhibit a marked decline in performance, underscoring the inherent difficulty of modeling articulated objects from monocular video sequences characterized by significant occlusions.

Tables 3, 4 and Figure 2 further demonstrate the robustness and accuracy of our method on challenging benchmarks established by prior work. In both the Video2Articulation-S and Arti4D datasets, the camera motion is primarily quasi-static, rendering them less demanding than our proposed sequences. Consequently, while baseline methods exhibit improved performance on these benchmarks, our approach consistently maintains superior results. Notably, the Arti4D dataset contains significant artifacts and "holes" in the depth maps; despite these imperfections, our method remains robust to low-quality depth inputs.

We provide visualizations of the proxy superquadrics in the appendix (Sec. F).

4.5 Ablation Analysis

In Table 5, we report the results of an ablation study on the AiP-synth dataset to evaluate the contribution of each component in our framework. Each experiment is performed using three distinct seeds, and we report the results from the trial yielding the minimum overall loss. To assess the impact of the reconstruction loss, we systematically remove $\mathcal{L}_{\text{rec}}^{\mathcal{X}\to\mathcal{Q}}$, $\mathcal{L}_{\text{rec}}^{\mathcal{Q}\to\mathcal{X}}$, and their combination. The utility of the motion cues is validated by removing the scene flow loss. Finally, we investigate the influence of the regularization terms $\mathcal{L}_{\text{reg}}$ by ablating the sparsity and existence constraints for both primitives and parts, along with the velocity magnitude and entropy constraints.

We observe a significant degradation in joint parameter estimation when the reconstruction loss terms are removed, as these components ensure that the primitives accurately fit and represent the observed point cloud. Similarly the omission of the flow loss prevents the model from effectively identifying moving components, resulting in a marked decline in dynamic part segmentation. Finally, the regularization terms facilitate convergence toward a physically plausible configuration by encouraging the model to represent the observed geometry with a minimal number of primitives and parts.

Table 4: **Quantitative results on the Arti4D dataset.** Best results are in **bold**.

Joint Type	Method	Axis ↓ (°)	Position ↓ (cm)	Type Acc. ↑ (%)
Revolute	Artipoint (43)	24.96 ± 35.27	85.49 ± 76.82	75.0
	Ours	**4.77 ± 3.79**	**5.44 ± 2.00**	**100.0**
Prismatic	Artipoint (43)	28.84 ± 11.09	–	64.4
	Ours	**3.60 ± 1.19**	–	**100.0**

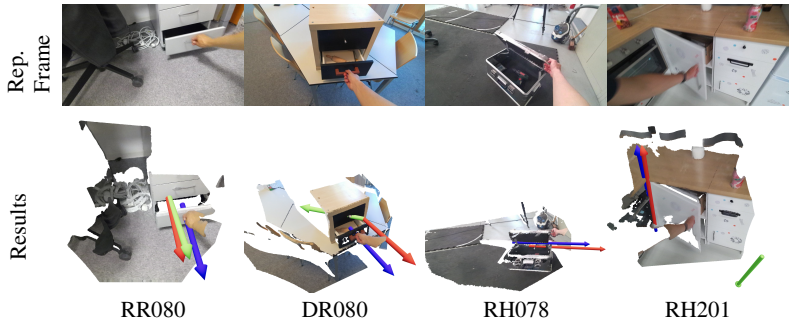

Figure 2: **Qualitative results on the Arti4D dataset.** Red, green, and blue arrows denote Ours, Artipoint, and Ground-Truth joint axes, respectively. For prismatic (translational) joints, only the axis direction is evaluated, while the axis location is not considered. Because Artipoint produces a large error in axis location for scene RH078, the green joint falls outside the camera frustum.

5 Limitations

Although we utilize superquadric primitives to obtain a coarse approximation of the object's geometry, this representation is insufficient for generating detailed mesh reconstructions. Consequently, incorporating more expressive primitive representations (9; 30) presents a promising avenue for future research, particularly for downstream applications like digital twins that necessitate high-fidelity surface reconstructions. Additionally, the proposed method is currently restricted to 1-DoF articulations with respect to a fixed base. Extending it to more complex articulated objects that require explicit kinematic tree prediction is left for future work.

6 Conclusion

We presented an unsupervised framework for discovering part labels and kinematic joint parameters of articulated objects. Extensive experiments across four challenging benchmarks demonstrate the accuracy and robustness of our approach, even when faced with complex camera trajectories, low-quality depth maps (e.g., in the Arti4D dataset), and non-standard objects like bottles or cameras. Furthermore, our per-video optimization formulation ensures strong generalization across diverse object categories.

Overall, our method is the first to frame articulated object reconstruction and joint parameter estimation as a primitive-fitting task. We demonstrate that casual monocular video provides sufficient geometric regularity to recover articulated motion without the need for dense correspondences or category-specific training.

Table 5: **Ablation Experiments**. We demonstrate the importance of each loss term used in our optimization. 'F' means the method fails to predict the metric. The complete method achieves state-of-the-art results across our evaluation suite. While specific sub-configurations may exhibit performance on par with the full method for certain metrics, they suffer from significantly higher error rates across the remaining criteria.

Method	mIoU ↑	Axis (°) ↓	Position (cm) ↓	Part Rotation (°) ↓	Part Translation (cm) ↓	Type Accuracy (%) ↑
w/o flow	0.26	13.62	F	F	3.62	75.0
w/o $\mathcal{L}_{\text{rec}}^{\mathcal{X}\to\mathcal{Q}}$	0.30	45.38	F	F	130.28	0.00
w/o $\mathcal{L}_{\text{rec}}^{\mathcal{Q}\to\mathcal{X}}$	0.57	22.50	0.02	3.5×10^{-3}	25.31	75.0
w/o $\mathcal{L}_{\text{rec}}^{\mathcal{X}\to\mathcal{Q}} + \mathcal{L}_{\text{rec}}^{\mathcal{Q}\to\mathcal{X}}$	0.34	44.74	F	F	48.00	25.0
w/o $\mathcal{L}_{\text{parts}}$	0.55	44.74	F	4.0×10^{-3}	19.78	50.0
w/o $\mathcal{L}_{\text{prims}}$	0.65	22.50	0.05	0.01	26.18	75.0
w/o $\mathcal{L}_{\text{match}}$	0.55	22.77	F	F	42.15	50.0
w/o $\mathcal{L}_{\text{motion}}$	0.58	45.00	F	13.76	39.27	50.0
Complete method	0.73	1.22×10^{-3}	4.1×10^{-3}	0.53	1.9×10^{-3}	100.0

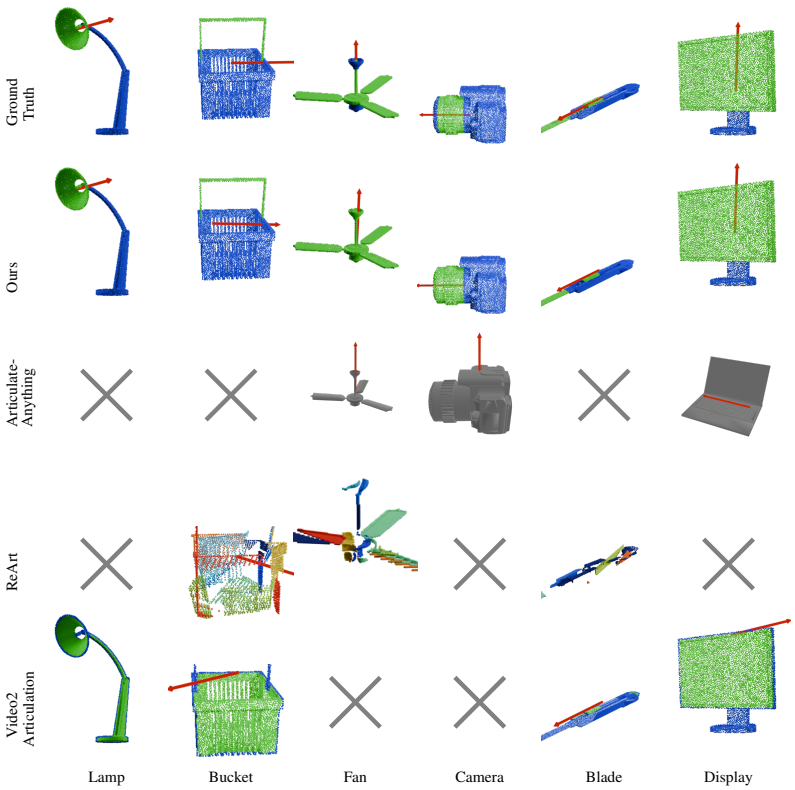

Figure 3: **Qualitative results on randomly selected sequences from the AiP-synth dataset.** Red arrows denote predicted joint axes. Note how our method retrieves the part segmentations and the joint axes much more accurately and robustly than all the other methods. '×' indicates that the method fails on the category. *Video results are provided in the video supplementary material.*

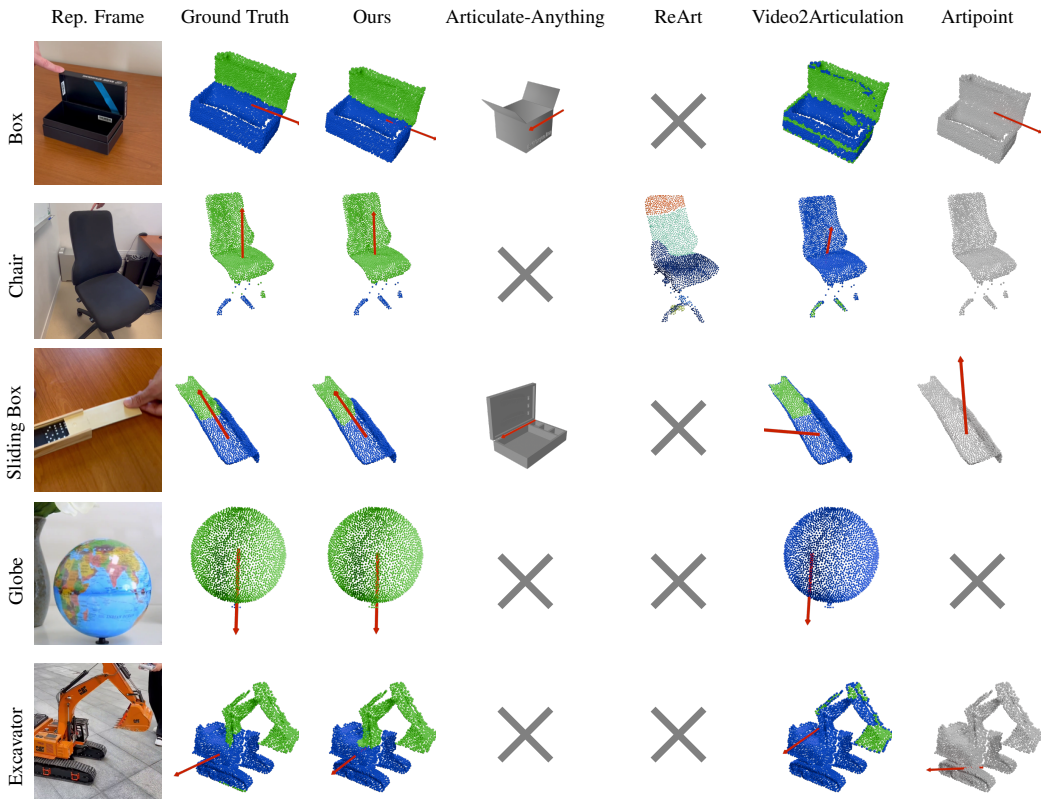

Figure 4: **Qualitative results on the real-world sequences from the AiP-real dataset.** Red arrows denote predicted joint axes. Note how our method retrieves the part segmentations and the joint axes much more accurately and robustly than all the other methods. '×' indicates that the method fails on the category. *Video results are provided in the video supplementary material.*

Acknowledgements

This project was funded in part by the European Union (ERC Advanced Grant explorer Funding ID #101097259).

References

[1] Ai, H., Chang, W., Jiao, J., Leonardis, A., Ofek, E.: Articulation in Motion: Prior-free Part Mobility Analysis for Articulated Objects By Dynamic-Static Disentanglement. In: International Conference for Learning Representations (2026)

[2] Artykov, A., Boittiaux, C., Lepetit, V.: Articulated Object Understanding from a Single Video Sequence. In: International Conference on Computer Vision Workshops (2025)

[3] Barr, A.H.: Superquadrics and angle-preserving transformations. IEEE Computer graphics and Applications **1**(1), 11–23 (1981)

[4] Chao, J.J., Jiang, Q., Isler, V.: Part Segmentation and Motion Estimation for Articulated Objects with Dynamic 3D Gaussians. In: British Machine Vision Conference (2025)

[5] Dai, H., Fan, H., Zhang, H., Wu, D., Zhang, J., Dong, H.: Freeartgs: Articulated gaussian splatting under free-moving scenario. arXiv preprint arXiv:2603.22102 (2026)

[6] Deng, B., Genova, K., Yazdani, S., Bouaziz, S., Hinton, G., Tagliasacchi, A.: Cvxnet: Learnable convex decomposition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 31–44 (2020)

[7] Fedele, E., Sun, B., Guibas, L., Pollefeys, M., Engelmann, F.: Superdec: 3d scene decomposition with superquadrics primitives. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 24625–24635 (2025)

[8] Ganeshan, A., Gadelha, M., Groueix, T., Chen, Z., Chaudhuri, S., Kim, V., Yifan, W., Ritchie, D.: Residual primitive fitting of 3d shapes with superfrusta. arXiv preprint arXiv:2512.09201 (2025)

[9] Groueix, T., Fisher, M., Kim, V.G., Russell, B.C., Aubry, M.: A papier-mâché approach to learning 3d surface generation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 216–224 (2018)

[10] Huang, J., Zhou, Q., Rabeti, H., Korovko, A., Ling, H., Ren, X., Shen, T., Gao, J., Slepichev, D., Lin, C.H., Ren, J., Xie, K., Biswas, J., Leal-Taixe, L., Fidler, S.: ViPE: Video Pose Engine for 3D Geometric Perception. In: arXiv Preprint (2025)

[11] Jiang, Z., Hsu, C.C., Zhu, Y.: Ditto: Building Digital Twins of Articulated Objects from Interaction. In: Conference on Computer Vision and Pattern Recognition (2022)

[12] Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3D Gaussian Splatting for Real-Time Radiance Field Rendering. In: ACM SIGGRAPH (2023)

[13] Kerr, J., Kim, C.M., Wu, M., Yi, B., Wang, Q., Goldberg, K., Kanazawa, A.: Robot See Robot Do: Imitating Articulated Object Manipulation with Monocular 4D Reconstruction. In: CORL (2024)

[14] Kim, S., Ha, J., Kim, Y.H., Lee, Y., Park, F.C.: Screwsplat: An end-to-end method for articulated object recognition. arXiv preprint arXiv:2508.02146 (2025)

[15] Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization 3rd international conference on learning representations. In: ICLR 2015-Conference Track Proceedings. vol. 1 (2015)

[16] Laine, S., Hellsten, J., Karras, T., Seol, Y., Lehtinen, J., Aila, T.: Modular primitives for high-performance differentiable rendering. ACM Transactions on Graphics **39**(6) (2020)

[17] Le, L., Xie, J., Liang, W., Wang, H.J., Yang, Y., Ma, Y.J., Vedder, K., Krishna, A., Jayaraman, D., Eaton, E.: Articulate-Anything: Automatic Modeling of Articulated Objects via a Vision-Language Foundation Model. In: International Conference for Learning Representations (2024)

[18] Li, X., Wang, H., Yi, L., Guibas, L.J., Abbott, A.L., Song, S.: Category-Level Articulated Object Pose Estimation. In: Conference on Computer Vision and Pattern Recognition (2020)

[19] Liu, J., Mahdavi-Amiri, A., Savva, M.: Paris: Part-Level Reconstruction and Motion Analysis for Articulated Objects. In: International Conference on Computer Vision (2023)

[20] Liu, S., Gupta, S., Wang, S.: Building Rearticulable Models for Arbitrary 3D Objects from 4D Point Clouds. In: Conference on Computer Vision and Pattern Recognition (2023)

[21] Liu, W., Wu, Y., Ruan, S., Chirikjian, G.S.: Robust and accurate superquadric recovery: A probabilistic approach. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2676–2685 (2022)

[22] Liu, W., Wu, Y., Ruan, S., Chirikjian, G.S.: Marching-primitives: Shape abstraction from signed distance function. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8771–8780 (2023)

[23] Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. In: ECCV (2020)

[24] Mo, K., Zhu, S., Chang, A.X., Yi, L., Tripathi, S., Guibas, L.J., Su, H.: PartNet: A Large-Scale Benchmark for Fine-Grained and Hierarchical Part-Level 3D Object Understanding. In: Conference on Computer Vision and Pattern Recognition (2019)

[25] Monnier, T., Austin, J., Kanazawa, A., Efros, A., Aubry, M.: Differentiable blocks world: Qualitative 3d decomposition by rendering primitives. Advances in Neural Information Processing Systems **36**, 5791–5807 (2023)

[26] Mu, J., Qiu, W., Kortylewski, A., Yuille, A., Vasconcelos, N., Wang, X.: A-SDF: Learning Disentangled Signed Distance Functions for Articulated Shape Representation. In: International Conference on Computer Vision (2021)

[27] Noguchi, A., Iqbal, U., Tremblay, J., Harada, T., Gallo, O.: Watch It Move: Unsupervised Discovery of 3D Joints for Re-Posing of Articulated Objects. In: Conference on Computer Vision and Pattern Recognition (2022)

[28] Ota, K., Tung, H.Y., Smith, K.A., Cherian, A., Marks, T.K., Sullivan, A., Kanezaki, A., Tenenbaum, J.B.: H-SAUR: Hypothesize, Simulate, Act, Update, and Repeat for Understanding Object Articulations from Interactions. In: International Conference on Robotics and Automation (2023)

[29] Park, J.J., Florence, P., Straub, J., Newcombe, R.A., Lovegrove, S.J.: DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation. In: Conference on Computer Vision and Pattern Recognition (2019)

[30] Paschalidou, D., Katharopoulos, A., Geiger, A., Fidler, S.: Neural parts: Learning expressive 3d shape abstractions with invertible neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3204–3215 (2021)

[31] Paschalidou, D., Ulusoy, A.O., Geiger, A.: Superquadrics revisited: Learning 3d shape parsing beyond cuboids. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10344–10353 (2019)

[32] Peng, W., Lv, J., Lu, C., Savva, M.: iTACO: Interactable Digital Twins of Articulated Objects from Casually Captured RGBD Videos. In: International Conference on 3D Vision (2026)

[33] Pilu, M., Fisher, R.B.: Equal-distance sampling of superellipse models (1995)

[34] Ravi, N., Gabeur, V., Hu, Y.T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., Mintun, E., Pan, J., Alwala, K.V., Carion, N., Wu, C.Y., Girshick, R., Dollár, P., Feichtenhofer, C.: SAM 2: Segment Anything in Images and Videos. In: arXiv Preprint (2024)

[35] Shi, Y., Cao, X., Lu, F., Zhou, B.: P^3-Net: Part Mobility Parsing from Point Cloud Sequences via Learning Explicit Point Correspondence. In: American Association for Artificial Intelligence Conference (2022)

[36] Shi, Y., Cao, X., Zhou, B.: Self-Supervised Learning of Part Mobility from Point Cloud Sequence. In: Computer Graphics Forum (2021)

[37] Song, C., Wei, J., Foo, C.S., Lin, G., Liu, F.: REACTO: Reconstructing Articulated Objects from a Single Video. In: Conference on Computer Vision and Pattern Recognition (2024)

[38] Tulsiani, S., Su, H., Guibas, L.J., Efros, A.A., Malik, J.: Learning shape abstractions by assembling volumetric primitives. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2635–2643 (2017)

[39] Vaskevicius, N., Birk, A.: Revisiting superquadric fitting: A numerically stable formulation. IEEE Transactions on Pattern Analysis and Machine Intelligence **41**(1), 220–233 (2019)

[40] Wang, H., Sridhar, S., Huang, J., Valentin, J., Song, S., Guibas, L.J.: Normalized Object Coordinate Space for Category-Level 6D Object Pose and Size Estimation. In: Conference on Computer Vision and Pattern Recognition (2019)

[41] Weng, Y., Wang, H., Zhou, Q., Qin, Y., Duan, Y., Fan, Q., Chen, B., Su, H., Guibas, L.J.: CAPTRA: CAtegory-level Pose Tracking for Rigid and Articulated Objects from Point Clouds. In: International Conference on Computer Vision (2021)

[42] Weng, Y., Wen, B., Tremblay, J., Blukis, V., Fox, D., Guibas, L., Birchfield, S.: Neural Implicit Representation for Building Digital Twins of Unknown Articulated Objects. In: Conference on Computer Vision and Pattern Recognition (2024)

[43] Werby, A., Buechner, M., Roefer, A., Huang, C., Burgard, W., Valada, A.: Articulated Object Estimation in the Wild. In: CoRL (2025)

[44] Wu, D., Liu, L., Linli, Z., Huang, A., Song, L., Yu, Q., Wu, Q., Lu, C.: Reartgs: Reconstructing and Generating Articulated Objects via 3D Gaussian Splatting with Geometric and Motion Constraints. In: Advances in Neural Information Processing Systems (2025)

[45] Xiang, F., Qin, Y., Mo, K., Xia, Y., Zhu, H., Liu, F., Liu, M., Jiang, H., Yuan, Y., Wang, H., Yi, L., Chang, A.X., Guibas, L.J., Su, H.: SAPIEN: A SimulAted Part-Based Interactive ENvironment. In: Conference on Computer Vision and Pattern Recognition (2020)

[46] Yang, K., Chen, X.: Unsupervised learning for cuboid shape abstraction via joint segmentation from point clouds. ACM Transactions On Graphics (TOG) **40**(4), 1–11 (2021)

[47] Zhang, Y., Edstedt, J., Wandt, B., Forssén, P.E., Magnusson, M., Felsberg, M.: GMSF: Global Matching Scene Flow. In: Advances in Neural Information Processing Systems (2023)

[48] Zhong, J.X., Cheng, T.Y., He, Y., Lu, K., Zhou, K., Markham, A., Trigoni, N.: Multi-Body SE(3) Equivariance for Unsupervised Rigid Segmentation and Motion Estimation. In: Advances in Neural Information Processing Systems (2023)

[49] Zhou, Y., Barnes, C., Lu, J., Yang, J., Li, H.: On the continuity of rotation representations in neural networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5745–5753 (2019)

Appendix

Articulation in Prime: Primitive-Based Articulated Object Understanding from a Single Casual Video

A Superquadrics

Superquadrics (3) form a parametric family of 3D shapes. Their popularity stems from their ability to represent a wide variety of shape variations with only a small number of parameters. A canonical superquadric is characterized by its scale $\mathbf{s} = (s_x, s_y, s_z) \in \boldsymbol{R}^3$ and its shape parameters $\boldsymbol{\varepsilon} = (\varepsilon_1, \varepsilon_2) \in \mathbb{R}^2$. It is common practice to constrain the shape parameters to the interval $[0.1, 1.9]$ in order to ensure convex superquadrics without overly sharp edges (39). Figure 5 illustrates how the shape parameters affect the geometry of superquadrics.

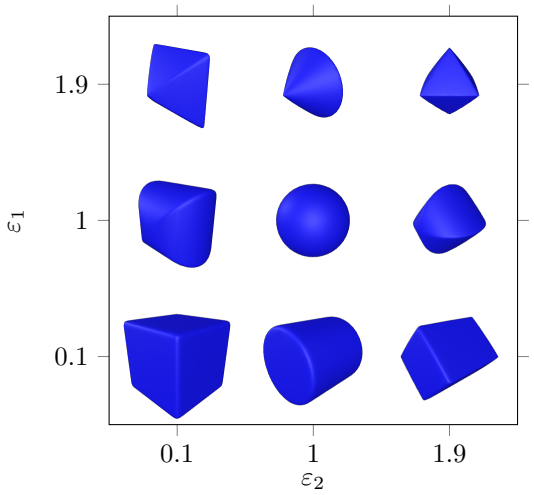

Figure 5: **Influence of the shape parameters on a unit scale superquadric.** The shape parameters are constrained to the interval $[0.1, 1.9]$ so that superquadrics are convex and more representative of shapes encountered in the real world.

A.1 Equations

The superquadric surface is defined by the implicit equation:

$$\left[\left(\frac{x}{s_x}\right)^{2/\varepsilon_2} + \left(\frac{y}{s_y}\right)^{2/\varepsilon_2}\right]^{\varepsilon_2/\varepsilon_1} + \left(\frac{z}{s_z}\right)^{2/\varepsilon_1} = 1 \tag{20}$$

or equivalently, by the explicit equation:

$$\begin{cases} x = s_x \cos^{\varepsilon_1} \eta \cos^{\varepsilon_2} \omega \\ y = s_y \cos^{\varepsilon_1} \eta \sin^{\varepsilon_2} \omega \\ z = s_z \sin^{\varepsilon_1} \eta \end{cases} \quad \begin{array}{l} \eta \in [-\pi/2,\, \pi/2] \\ \omega \in [-\pi,\, \pi] \end{array} \tag{21}$$

A.2 Point Sampling

The implicit equation 20 proves useful for sampling points on the surface of a superquadric. Let $\boldsymbol{v} = (v_x, v_y, v_z) \in \mathbb{R}^3$ be a random direction satisfying $\|\boldsymbol{v}\|_2 = 1$. Solving for $t \geq 0$ such that $(x, y, z) = t\boldsymbol{v}$ lies on the surface of the unit superquadric yields:

$$t = \left[\left(|v_x|^{2/\varepsilon_2} + |v_y|^{2/\varepsilon_2}\right)^{\varepsilon_2/\varepsilon_1} + |v_z|^{2/\varepsilon_1}\right]^{-\varepsilon_1/2} \tag{22}$$

We then apply the scaling parameters to obtain the corresponding point on the surface of the scaled superquadric. This simple sampling procedure is differentiable by construction, fast to compute, and empirically performs on par with the standard uniform sampling method by Pilu *et al.* (33).

A.3 Mesh Construction

A mesh representation of a superquadric can be easily derived from the explicit equation 21, following (25). The idea is to define a mapping between the unit sphere and the scaled superquadric, the sphere being a special case of the superquadrics with shape parameters $\boldsymbol{\varepsilon} = (1, 1)$ (Figure 5). We first construct a unit icosphere whose vertices all lie on the unit sphere. Then, for each vertex, we compute the pair (η, ω) using the explicit equation of the unit sphere, and apply the explicit equation of the scaled superquadric to get the deformed point. The mesh connectivity is preserved.

B Implementation Details

B.1 Design Choices

In our experiments, we optimize over sequences of $T = 25$ frames, sampling a point cloud of $N = 4096$ point from each frame. The maximum number of primitives that can be used in the reconstruction is set to $Q = 10$, while the maximum number of parts is set to $P = 8$. Superquadrics are initialized with a uniform scale of 0.5, shape parameters $(\varepsilon_1, \varepsilon_2) = (0.2, 0.2)$ (close to cuboids), an existence probability of 0.5, and a random pose at the first timestep. We use the 6D rotation representation (49) to parameterize the orientation at the first timestep.

We use a 3-layer MLP to compute point features from the 3D coordinates. Both point and primitive feature vectors are of dimension 16.

B.2 Optimization Details

To compute the reconstruction loss, we sample 500 points per primitive using the procedure described in Section A.2. Occlusions are computed efficiently by rendering the primitives into a Z-buffer with Nvdiffrast (16), where superquadric meshes are derived from transformed icospheres as described in Section A.3.

We optimize for 10k steps with a learning rate of 5×10^{-3}, without learning rate warmup of scheduling, and employ the Adam (15) optimizer.

Table 6 reports the weighing coefficients of our loss terms.

Loss term	Weight
$\mathcal{L}_{rec}^{\mathcal{Q}\to\mathcal{X}}$	2.0
$\mathcal{L}_{rec}^{\mathcal{X}\to\mathcal{Q}}$	2.0
$\mathcal{L}_{\text{flow}}$	1.0
$\mathcal{L}_{\text{part-sparsity}}$	0.1
$\mathcal{L}_{\text{part-existence}}$	0.05
$\mathcal{L}_{\text{prim-sparsity}}$	0.01
$\mathcal{L}_{\text{prim-existence}}$	0.05
$\mathcal{L}_{\text{motion}}$	0.1
$\mathcal{L}_{\text{match}}$	0.1

Table 6: **Weighting factors of our loss terms.** These weights were set manually with minimal tuning.

B.3 Computational Cost

Our method runs in about 10 minutes per object on a single GPU NVIDIA A6000 with a processor 32x Xeon(R) Silver 4208. We run 5 seeds per object and pick the result with lowest loss after the optimization converges. For the ablations we run only 3 seeds per object.

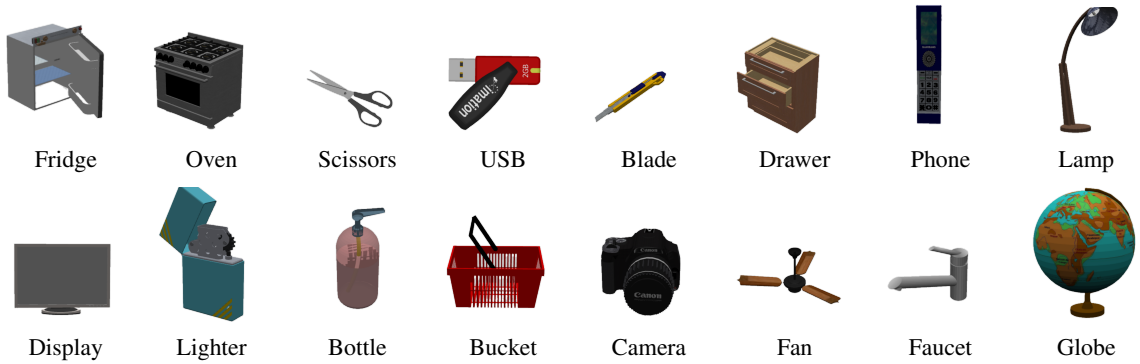

Figure 6: **Representative frames of all the objects used in our AiP-synth dataset.** This selection covers a wide range of geometries and kinematics. Some object parts are particularly complex and require the use of multiple primitives. Other objects, such as the globe, exhibit symmetries that make the joint discovery more challenging.

C Details on AiP-synth and AiP-real Datasets

C.1 AiP-synth

The dataset comprises a diverse selection of objects extracted from PartNet-Mobility, illustrated in Figure 6. To simulate a real capture system, we define a circular trajectory and a ping-pong trajectory. We illustrate them in Figure 7.

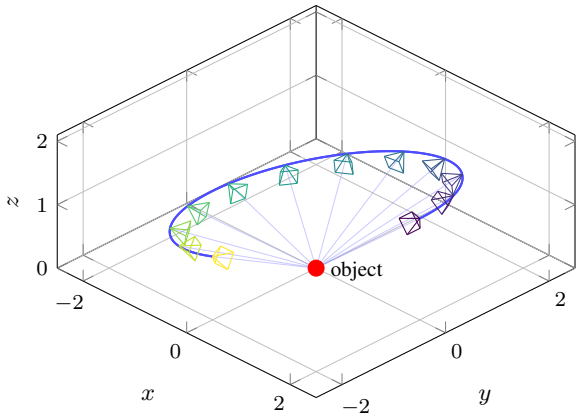

Figure 7: **A synthetic circular camera trajectory used in AiP-synth.** The camera turns around the object with random start and end angles, and a random height variation amplitude. In the case of the ping-pong trajectory, the camera sweeps back and forth along the circular path.

Circular trajectory. Let $t \in [0, 1]$ parametrize the trajectory. The azimuth angle is linearly interpolated between the start and end angles:

$$\theta(t) = \theta_{\text{start}} + t\,(\theta_{\text{end}} - \theta_{\text{start}}) \tag{23}$$

Finally, the camera position $\mathbf{p}(t) \in \mathbb{R}^3$ is given by:

$$\mathbf{p}(t) = \begin{pmatrix} r\cos\theta(t) \\ r\sin\theta(t) \\ h_0 + \Delta h\cos\theta(t) \end{pmatrix} \tag{24}$$

where r is the radius, h_0 the base height, and Δh the height variation amplitude.

Table 7: **Stress-test experiment results on the AiP-synth dataset.**

Joint Type	Method	mIoU ↑	Axis ↓ (°)	Position ↓ (cm)	State ↓ (° / cm)
Revolute	with DA-v2	0.85 (0.06)	0.82 (0.50)	8.68 (8.31)	1.17 (0.17)
	with ViPE pose	0.67 (0.24)	0.43 (0.11)	15.67 (15.30)	13.98 (12.98)
	with GMSF flow	0.66 (0.25)	2.84 (2.52)	16.01 (15.64)	11.33 (10.33)
	with DA-v2, ViPE pose, and GMSF	0.60 (0.31)	14.90 (14.58)	17.81 (17.44)	24.31 (23.31)
	Ours	0.91	0.32	0.37	1.00
Prismatic	with DA-v2	0.58 (0.1)	2.23 (2.13)	–	7.75 (6.19)
	with ViPE pose	0.53 (0.15)	2.91 (2.81)	–	22.63 (21.07)
	with GMSF flow	0.41 (0.27)	2.68 (2.58)	–	18.62 (17.06)
	with DA-v2, ViPE pose, and GMSF	0.27 (0.41)	10.38 (10.28)	–	26.45 (24.89)
	Ours	0.68	0.1	–	1.56
Mean Depth Error	Abs Rel = 0.02				
Mean Camera Pose Error	ATE RMSE = 0.44	Rot err = 5.0°			
Flow Error	EPE = 0.0113	EPE RMSE = 0.0153	EPE Median = 0.0091		

Ping-pong trajectory. We define the oscillation fraction $f(t) \in [0, 1]$ as:

$$f(t) = \begin{cases} \dfrac{1 - \cos(2\pi t)}{2} & \text{(smooth turnaround)} \\ 1 - 2\left|(t - \lfloor t \rfloor) - \frac{1}{2}\right| & \text{(constant speed)} \end{cases} \tag{25}$$

The azimuth angle is then interpolated:

$$\theta(t) = \theta_{\text{start}} + f(t)\,(\theta_{\text{end}} - \theta_{\text{start}}) \tag{26}$$

and the camera position still follows Equation 24. The camera now performs a single back and forth sweep along a circular arc.

C.2 AiP-real

Within AiP-real, the `globe` and `excavator` sequences are extracted from YouTube videos. The strong performance of our method on these sequences highlights its practicality and robustness in an in-the-wild setting.

Ground-truth labels are difficult to obtain on real-world objects. We leveraged SAM2 (34) to assist with segmentation annotations. Annotating a joint axis is significantly more ambiguous for a human. To address this, we leveraged our physical understanding of the objects: we manually selected two points lying on the joint axis from the images, and back-projected them to 3D using the depth maps to recover the 3D axis.

D Additional Experiments

D.1 Robustness Experiments

To evaluate the resilience of our framework against noisy inputs from off-the-shelf estimators, we conducted stress-test experiments across the 6 representative categories. Specifically, we re-executed AiP-synth using depth maps from Depth-Anything-v2, camera trajectories from ViPE, scene flow from GMSF, and a joint configuration incorporating all three estimated modalities. Table 7 reports the results averaged across all random seeds, with values in parentheses denoting the absolute performance drop. We also provide the corresponding depth, pose, and flow estimation errors averaged across all frames and categories.

While replacing ground-truth inputs with upstream estimates predictably lowers accuracy, our optimization framework provides strong geometric regularization against these errors (Table 7). We highlight several key findings:

Depth (DA-v2) introduces temporal surface jitter ("waving flag" artifacts). Nonetheless, part segmentation remains stable—with Revolute mIoU decreasing by only 0.06—as the volumetric superquadric prior effectively filters out high-frequency surface noise. **Camera Pose (ViPE)** induces inter-frame translation drift, severely impacting kinematic state estimation (e.g., a 21.07 reduction in Prismatic state). This stems from the ambiguity in separating extrinsic camera motion from intrinsic articulation

during optimization. **Scene Flow (GMSF)** exhibits spurious motion on static regions, degrading temporal tracking and lowering segmentation quality (a 0.25 drop in Revolute mIoU). In the **Joint Setting**, although compounded errors degrade performance further (Revolute mIoU reaches 0.60), the pipeline avoids catastrophic failure, highlighting the structural robustness provided by our superquadric representation.

Crucially, running these off-the-shelf estimators on synthetic images introduces a severe domain gap. This inherently degrades their initial predictions, artificially lowering the final metrics independently of our method's true performance.

Furthermore, our evaluation on the Arti4D dataset on Table 4 and Figure 2 demonstrates the method's robustness to alternative input modalities (e.g., depth sensors and camera odometry) beyond monocular predictors.

D.2 Loss vs. Metric Correlation Analysis

To verify that our loss is an effective model selection criterion across multiple runs, Figure 8 plots final loss against mIoU across 5 random seeds for the representative categories. We observe a strong negative correlation, where higher loss systematically reflects lower segmentation accuracy. In all tested categories, identifying the run with the minimal loss uniquely recovers the peak mIoU, demonstrating that our optimization objective is an effective surrogate for end-task performance.

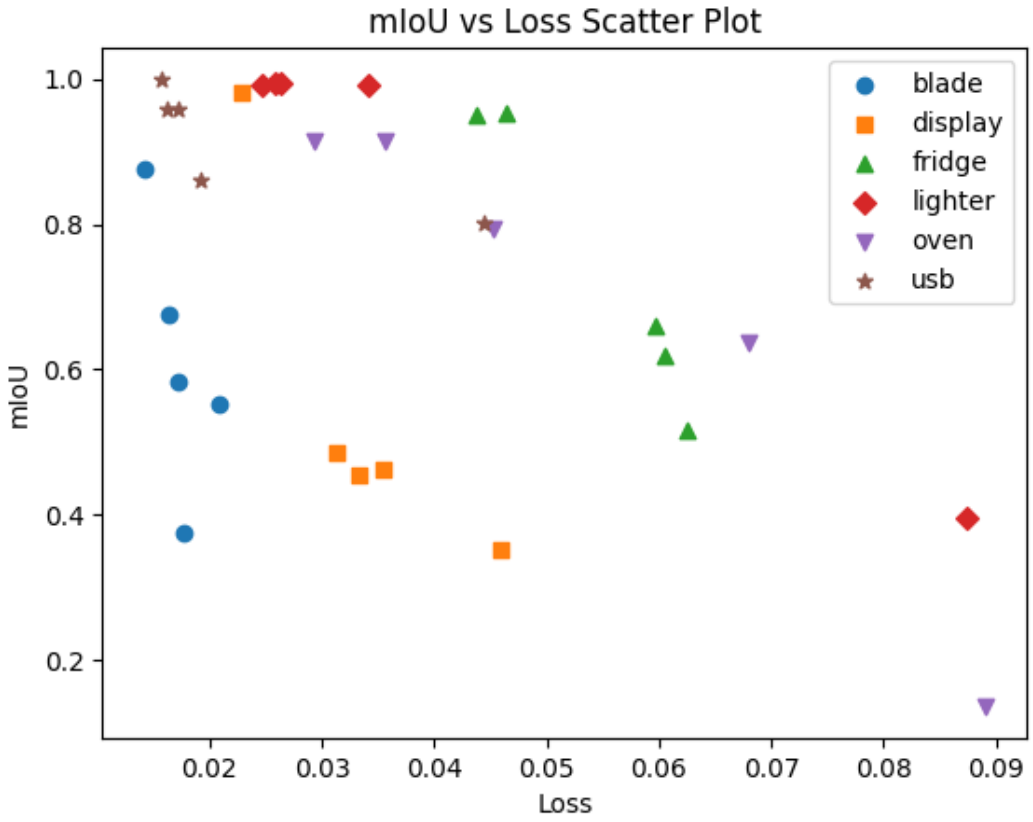

Figure 8: **mIoU vs Loss Metric Scatter Plot.** The results demonstrate a strict negative correlation: higher optimization loss reliably indicates a degradation in segmentation accuracy. Most importantly, for every category tested, selecting the seed with the minimum optimization loss successfully isolates the run with the highest mIoU, validating our loss formulation as a robust and mathematically sound selection signal.

E Additional Qualitative Results on AiP-synth

Figure 9 shows qualitative results on the remaining sequences of the AiP-synth dataset. Our approach again yields better results than competing methods.

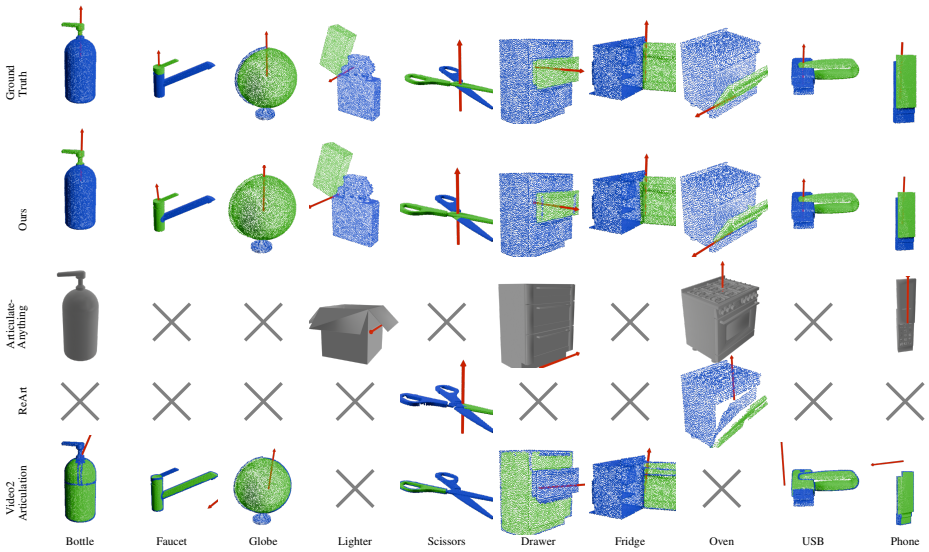

Figure 9: **Qualitative results on the rest of the sequences from the AiP-synth dataset.** Red arrows denote predicted joint axes. Note how our method retrieves the part segmentations and the joint axes much more accurately and robustly than all the other methods.

F Visualization of Object Decompositions into Primitives and Parts

Beyond serving as proxies to group object points into geometrically and dynamically consistent regions, primitives also provide interpretability. We present primitive decompositions for AiP-synth and AiP-real objects in Figures 10 and 11, respectively. We observe that certain parts automatically aggregate multiple primitives to reconstruct complex object structures beyond the modeling capacity of a single superquadric, such as the lamp base, the chair body or the excavator arm. Despite their limited expressivity, which prevents faithful reconstruction of real-world objects shapes, they prove sufficient for part segmentation and articulation discovery, supporting our design choice.

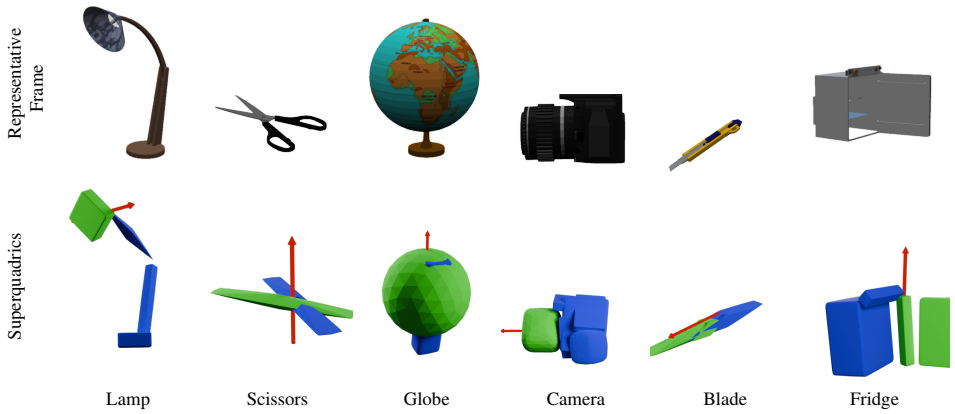

Figure 10: **Fitted superquadrics results on sequences from the AiP-synth dataset.** Red arrows denote predicted joint axes.

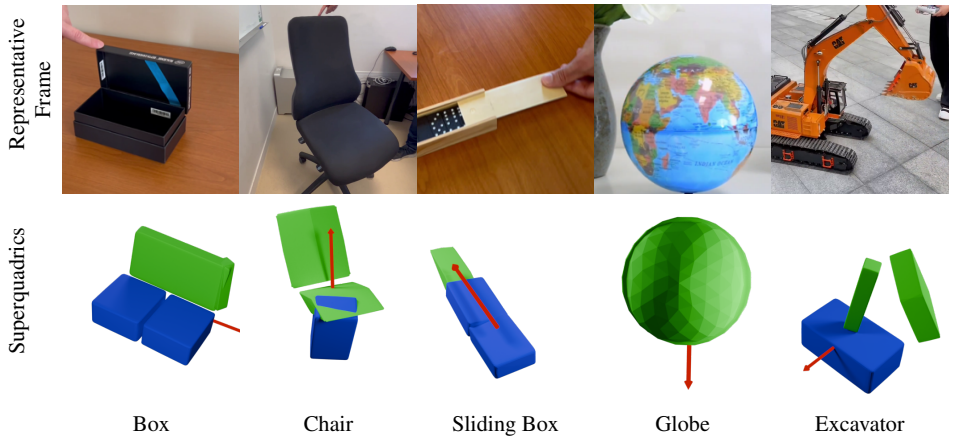

Figure 11: **Fitted superquadrics results on the sequences from the AiP-real dataset.** Red arrows denote predicted joint axes.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: In the context of articulated object understanding, our abstract and introduction present our primitive-based optimization approach that uses superquadric primitives as a proxy representation to predict part segmentation into static and dynamic and articulation parameters. In both sections the main limitations of previous work and our contributions are presented.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We have included a Limitations section discussing the main limitations of our approach.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: Our approach does not involve theoretical mathematical proofs, thus Theorems and Lemmas are not included in the paper.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We have included detailed assumptions made in order to ensure reproducibility of our results. A more detailed implementation description is included in the supplemental.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Our code and data will be made publicly available, and scripts to reproduce the results of the paper will be provided.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: All the parameters of our approach, experiments and ablations are included in the paper, and the supplemental material includes extra resutls and details.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Our experiments, in Table 1,3, and 4 report average results across the dataset with standard deviation. For individual object results (Table 2) we report best results based on lowest loss on a set of 5 independent runs, and for ablations (Table 5) we only report average results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We have included details on the machine used to run our experiments in the the supplemental.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our research conforms to NeurIPS guidelines.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: Our paper does not pose a risk of societal impacts.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: Our paper does not pose such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Data from previous work is properly referenced.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: New data introduced in the paper is captured either by the authors or retrieved from publicly available sources.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: Our method did not involve a study with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: Our method did not involve a study with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: Our method does not involve usage of LLMs.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.