ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 979-8-3503-6874-1/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICASSP49660.2025.10889272

Towards Robust Category-level Articulation Pose Estimation via Integrated Differentiable Rendering

Xinyi Yu*, Haonan Jiang*, Li Zhang†, Yukang Huo §, Lin Yuanbo Wu ‡, Yanyan Wei¶, Rohit Agarwal ‖, Harshal Suresh Shende**, Liu Liu¶, and Linlin Ou *
*Zhejiang University of Technology, China, † University of Science and Technology of China, China, §China Agricultural University, China, ‡Swansea University, UK, ¶Hefei University of Technology, China, ‖Department of Computer Science and Engineering, National Institute of Technology, Durgapur, India, **Indian Institute Of Technology, Indian
Email: {yuxy, 211122030127, linlinou}@zjut.edu.cn; liuliu@hfut.edu.cn; zanly20@mail.ustc.edu.cn

***Abstract*—Accurate object pose estimation is crucial for embodied intelligence tasks such as manipulation, grasping, and human-robot interaction. However, due to the inherent characteristics of articulated objects, such as kinematic constraints and self-occlusion, pose estimation for articulated objects has remained a significant challenge. To address these issues, this paper proposes CAPED, an end-to-end robust Category-level Articulated object Pose Estimator integrated differentiable rendering. Given partial point cloud as input, CAPED outputs the per-part 6D pose for articulation. Specifically, with the proposed joint-centric modeling manner, CAPED firstly estimates the pose for the free part. Afterward, we canonicalize the input point cloud to estimate constrained parts' poses by predicting the joint parameters and states as replacements. For further refinement, we propose a differentiable rendering scheme for pose optimization. Evaluations of the ArtImage and RobotArm datasets demonstrate that CAPED exhibits outstanding effectiveness and generalization in tasks ranging from synthetic data to real-world scenarios. We will publicly release the code.**

***Index Terms*—Articulated Objects, Pose Estimation, Differentiable Rendering.**

I. Introduction

Articulated objects [1]–[3], are ubiquitous in our daily lives, ranging from small items like eyeglasses to large appliances like dishwashers. Unlike rigid objects [4]–[7], which are viewed as single entities in three-dimensional space, articulated objects consist of multiple movable rigid parts connected by joints, following specific motion structures. Accurate pose estimation for both categories of objects is a critical task in numerous computer vision and robotics applications, yet this area has not been fully explored, particularly in applications such as augmented reality [8]–[11], 3D scene understanding [12]–[14], and robotic manipulation [15]–[20]. Despite significant advancements in recent years, articulated object pose estimation continues to face two major challenges:

(a)Self-occlusion. Previous methods [21], [22] lack robustness in handling self-occlusion, particularly when larger parts occlude smaller movable parts from some camera viewpoints.

(b)Inaccurate performance. Compared to 3D point clouds, 2D mask images offer higher information aggregation, enabling better capture of subtle pose variations during optimization. However, existing SOTA methods [21], [23], [24] tend to neglect this advantage, limiting pose estimation performance.

Corresponding authors: Linlin Ou and Liu Liu.

In this paper, we introduce a novel method for articulated object modeling, along with a customized refinement. To address the first challenge, we propose joint-centric modeling. This involves canonicalizing the pose of the ***free*** part by transforming the input point cloud (PC) from camera space to canonical space. In canonical space, we estimate the poses of ***constrained*** parts by predicting joint parameters and states. To tackle the second challenge, we implement a differentiable rendering process. Specifically, we found that 2D mask images play a crucial role. However, the conventional binarization rendering function can't be directly applied during training. Our key contribution is making the rendering process end-to-end trainable, integrating it seamlessly into the optimization process. All in all, our key contributions are threefold:

- CAPED is a novel framework aiming to solve the problem of category-level articulated object 6D pose estimation, where the pose estimation problem is cast into pose optimization task in canonical space.
- To address challenges such as self-occlusion and pose optimization, CAPED introduces modules such as joint-centric pose modeling and differentiable rendering.
- The efficacy and robustness of CAPED are demonstrated through evaluation on both point cloud observations and RGB-D images, using datasets ranging from synthetic to real-world scenarios.

II. Methodology

Problem Statement and Notations. Shortly, given the partial observation $\mathcal{P} \in \mathbb{R}^3$ with K rigid parts as **input**, our CAPED takes per-part rotation, translation, and scale $\{R^{(k)}, \boldsymbol{t}^{(k)}, \boldsymbol{S}^{(k)}\}_{k=1}^{K}$ as **output**, where $\{R^{(k)}\}_{k=1}^{K} = \{R_{\textbf{free}}, \{R_{\textbf{cons}}^{(k)}\}_{k=2}^{K}\}$, $\{\boldsymbol{t}^{(k)}\}_{k=1}^{K} = \{\boldsymbol{t}_{\textbf{free}}, \{\boldsymbol{t}_{\textbf{cons}}^{(k)}\}_{k=2}^{K}\}$.

Network Design. We adopt HS-Encoder [25] as backbone, due to its ability to perceive both local and global geometric information and its robustness to noise. All decoders are composed of multiple one-dimensional convolution blocks.

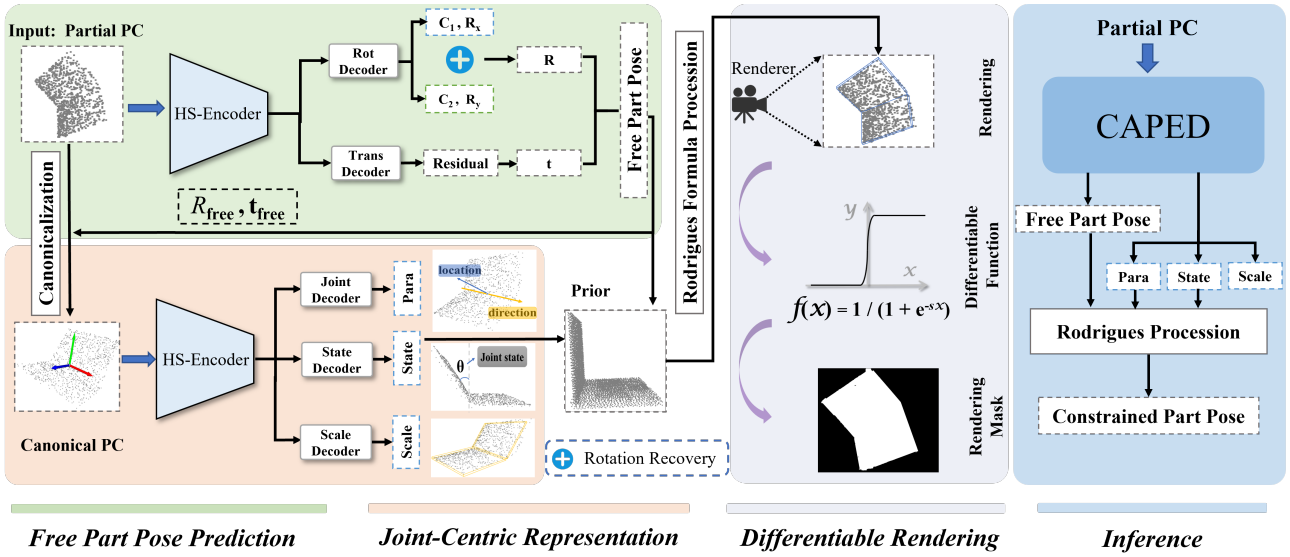

Fig. 1: **The pipeline of our CAPED**. Given the partial PC as input, our CAPED output per part pose for articulation.

A. Decoupled Pose Estimation

Inspired by GPV-Pose [26], the R_{free} can be decomposed into three direction vectors: x, y, and z, we predict two of them via HS-Encoder [25], achieving lower complexity. An attempt is made to estimate a confidence value for each vector [26] to enhance the robustness during the recovery of the final 3D rotation. Regarding the translation $\boldsymbol{t}$, we firstly zero-center the input point cloud $\mathcal{P}$ by reducing the center coordinate of point cloud $\widetilde{\mathcal{P}}$. For translation $\boldsymbol{t}_{\text{free}}$, we predict the residual value $\boldsymbol{t}^*_{\text{free}}$ for better prediction through an MLP. The final translation of free part $\boldsymbol{t}_{\text{free}}$ can be obtained by $\boldsymbol{t}_{\text{free}} = \boldsymbol{t}^*_{\text{free}} + \widetilde{\mathcal{P}}$. Finally, the 6D pose of the free part is the constitution of rotation and translation $T_{\text{free}} = \{R_{\text{free}}, \boldsymbol{t}_{\text{free}}\}$.

B. Joint-centric Representation of Articulation

A common approach for articulated objects is part-centric representation, treating them as combinations of rigid objects, as widely used in previous works [21], [22]. While straightforward, this method neglects kinematic relationships and faces challenges with self-occlusion. In contrast, we adopt a joint-centric perspective to model articulated objects.

We categorize rigid parts of an articulated object into two groups: free parts, which move arbitrarily in camera space, and constrained parts, which move according to joint constraints. With joint parameters, motion is determined by joint type. Following A-NCSH [21], we consider: (1) Revolute joints, allowing rotational motion, with joint state $\boldsymbol{\theta}_r$ defined by the rotation angle and parameters $\boldsymbol{\phi}_r = (\boldsymbol{u}_r, \boldsymbol{q}_r)$, where $\boldsymbol{u}_r$ is the joint axis and $\boldsymbol{q}_r$ is the pivot point. (2) Prismatic joints, allowing translational motion, with joint state $\boldsymbol{\theta}_p$ and parameters $\boldsymbol{\phi}_p = (\boldsymbol{u}_p)$. Due to point cloud can provide both a shape prior and a kinematic prior in canonical space [27], we canonicalize the input point cloud as follows:

$$\hat{\mathcal{P}}^{(k)} = (T_{\text{free}})^{-1}\mathcal{P}^{(k)} = (R_{\text{free}})^{-1}(\mathcal{P}^{(k)} - \boldsymbol{t}_{\text{free}}) \tag{1}$$

where $\hat{\mathcal{P}}^{(k)}$ denotes canonical point cloud. With the predicted joint state and parameter in canonical space, the Rodrigues formula is used to convert joint states into a matrix $R'^{(k)}_{\text{cons}}$ for the k-th part.

$$\begin{aligned} R'^{(k)}_{\text{cons}} = &\cos\boldsymbol{\theta}_r^{(k)} U_r^{(k)} + (1 - \cos\boldsymbol{\theta}_r^{(k)}) \cdot \\ &(U_r^{(k)} \cdot Q_r^{(k)}) Q_r^{(k)} + \sin\boldsymbol{\theta}_r^{(k)} (Q_r^{(k)})^{\wedge} \end{aligned} \tag{2}$$

where $U_r^{(k)}$ denotes the normalized direction of the joint axis $\boldsymbol{u}_r^{(k)}$, $Q_r^{(k)}$ denotes a anti-symmetric matrix composed of pivot point position $\boldsymbol{q}_r^{(k)}$. Here the $R'^{(k)}_{\text{cons}}$ is a matrix of size 3×4 and we stack it with the row [0, 0, 0, 1] appended at the end to get relative pose $T'^{(k)}_{\text{cons}}$ of k-th constrained part.

For prismatic joint, given predicted joint parameters $(\boldsymbol{u}_p^{(k)})$ and joint state $\boldsymbol{\theta}_p^{(k)}$, we can also get relative pose for k-th constrained part:

$$T'^{(k)}_{\text{cons}} = \begin{bmatrix} \mathbf{I} & \boldsymbol{\theta}_p^{(k)} \boldsymbol{u}_p^{(k)} \\ 0 & 1 \end{bmatrix} \tag{3}$$

where $\mathbf{I}$ indicates the identity matrix. Finally, we can calculate the poses of constrained parts by $T^{(k)}_{\text{cons}} = T_{\text{free}} T'^{(k)}_{\text{cons}}$. The total articulated object poses for all the K parts can be represented by a sequence of constrained part pose $\{T^{(k)}_{\text{cons}}\}_{k=2}^{K}$ and free part pose $T_{\text{free}} = \{R_{\text{free}}, \boldsymbol{t}_{\text{free}}\}$.

After obtaining the pose of free part and joint state, we use Eq. 2 and Eq. 3 to conduct point cloud prior transformation to get the transformed prior.

C. Differentiable Rendering

For further refinement, we employ a differentiable rendering process. Specifically, we employ a renderer that back-project the transformed prior of 3D object anchor into 2D mask with the pre-defined orthographic camera setting and a point rasterizer [28]. Here, the rendered mask is represented as $\mathcal{M}_0 \in \mathbb{R}^{H \times W}$, where H and W represent the height and width of the mask, respectively. The projection binarization process is typically formulated as $B_{i,j} = \mathbb{1}\{P_{i,j} \geq T\}$. Here, T is a hyperparameter (pixel threshold), (i, j) denotes the coordinates of the mask array, and $\mathbb{1}\{\cdot\}$ is the indicator

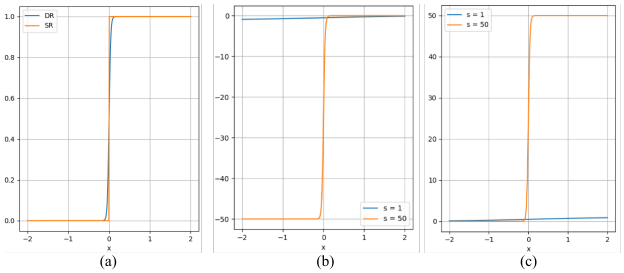

Fig. 2: **Illustration of differentiable binarization and its derivative.** (a) Numerical comparison of standard Rendering (SR) and differentiable Rendering (DR). (b) Derivative of ℓ_+. (c) Derivative of ℓ_-.

function. However, this equation cannot be directly applied during network training, as the standard binarization rendering process is non-differentiable. To address this issue, we draw inspiration from Differentiable Soft Quantization [29] and approximate the rendering process with a step function. Thus, the projection binarization can be re-formulated as:

$$\hat{B}_{i,j} = \frac{1}{1 + \mathrm{e}^{-s \cdot (P_{i,j} - \sigma T)}} \tag{4}$$

where $\hat{B}$ is the approximate binary map, T is the threshold. s indicates the scaling factor, which is set to 50 empirically. σ is the adaptive factor, inherently 1×1 convolution. In this way, the approximate binarization rendering function behaves similarly to the standard binarization rendering function (see Fig. 2) but is differentiable and thus can be optimized along with the network in the training period.

The performance improvement attributed to differentiable rendering (DR) can be explained through the backpropagation of gradients. Consider the binary cross-entropy loss as an example. We define the DR function as $f(x) = 1/1 + \mathrm{e}^{-sx}$, where $x = P_{i,j} - \sigma T$. Using this function, the losses ℓ_+ for positive labels and ℓ_- for negative labels can be readily derived. By applying the chain rule, the derivatives of these losses can be computed. The loss functions and their partial derivatives for positive labels are provided in Eq. 5, while those for negative labels are shown in Eq. 6.

$$\ell_+ = -\log(0 - \frac{1}{1 + e^{-sx}}), \qquad \frac{\partial \ell_+}{\partial x} = \frac{-s \cdot \mathrm{e}^{-s \cdot x}}{1 + e^{-s \cdot x}} \tag{5}$$

$$\ell_- = -\log(1 - \frac{1}{1 + e^{-sx}}), \qquad \frac{\partial \ell_-}{\partial x} = \frac{s}{1 + \mathrm{e}^{-s \cdot x}} \tag{6}$$

The derivatives of ℓ_+ and ℓ_- are also shown in Fig. 2. We can perceive from the differential that (1) The gradient is augmented by the amplifying factor s; (2) The amplification of gradient is significant for most of the wrongly predicted region ($x < 0$ for ℓ_+; $x > 0$ for ℓ_-), thus facilitating the optimization and helping to produce more distinctive predictions. Moreover, as $x = P_{i,j} - \sigma T$, the gradient of P is affected and rescaled between the foreground and the background by T.

D. Optimization and Inference

Loss Functions. The loss function consists of three parts 1) for rotation loss ℓ_{rot}, translation loss ℓ_{trans}, joint state loss ℓ_{sta} and scale loss ℓ_{scl}, we use ℓ_1 loss; 2) for joint parameters loss ℓ_{jnt}, we use ℓ_2 loss. 3) For rendered mask loss ℓ_{diff}, we use pixel-wise loss from Eq. 5 and Eq. 6. Finally, the overall loss function can be written as: $\mathcal{L} = \lambda_1 \ell_{rot} + \lambda_2 \ell_{trans} + \lambda_3 \ell_{sta} + \lambda_4 \ell_{scl} + \lambda_5 \ell_{jnt} + \lambda_6 \ell_{diff}$, where $\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6$ are 5.0, 5.0, 5.0, 3.0, 1.0, 1.0, empirically.

Inference. During inference, our CAPED directly outputs the pose of free part, scale, joint state and parameters. We apply the effective Rodrigues Procession to obtain constrained part pose without extra time-consuming post-process [21].

III. Experiments

A. Experimental Setting

We evaluate our CAPED framework on synthetic dataset ArtImage [23] and real dataset RobotArm [30]. ArtImage contains four categories with evolute joint: Laptop, Eyeglasses, Dishwasher, and Scissors; one category drawer with prismatic joint. To validate the generality of our method in real-world scenarios, we also train and test on the 7-part RobotArm dataset. A-NCSH [21], OMAD [23] and ContactArt [24] are the baselines. For validation, we adopt degree error ($^\circ$) for 3D rotation, distance error (m) for 3D translation, 3D IOU (%) for scale [21]. We also provide joint state metric since we use the joint-centric pose modeling method. For part-centric modeling, the joint state can be calculated from the poses between the parts [27] according to the kinematic relationship. We adopt a ranger optimizer, the number of total training epochs is 200, batch size is 16. All the experiments are implemented on an NVIDIA GeForce RTX 3090 GPU with 24GB memory.

B. Comparison with the SOTA Methods

Synthetic Dataset. We report the results of CAPED on ArtImage in Tab. I. Compared to the classical methods, we get the best pose estimation result on category *laptop*, with $\mathbf{3.5}^\circ, \mathbf{4.0}^\circ$ on rotation error. This can be explained by the proposed differentiable rendering strategy can outperform objects with similar size and shape at part level. For category Eyeglasses, our method achieves excellent performance with only $\mathbf{0.038}m$, $\mathbf{0.098}m$, and $\mathbf{0.089}m$ on translation error, which can be attributed to the framework's accurate estimation of state. Moving to 3D IoU metric, we see a significant improvement in predicting scale of each part compared to the baselines. Regarding joint state error, we achieve $\mathbf{2.0}^\circ$ on both laptop and diashwasher, showing direct regression in joint-centric perspective is more effective than calculation in part-centric perspective. Qualitative results are illustrated in Fig.3 (left).

Real-world Scenarios. To assess the generalization capability of our CAPED, we conducted experiments on real-world scenarios using the 7-part RobotArm dataset. The quantitative results are presented in Tab. II, achieving $\mathbf{16.7}^\circ$ on rotation error and $\mathbf{0.312}m$ on translation error for the last part. It shows that our method performs well on objects with diverse structures. Qualitative results are illustrated in Fig.3 (right).

C. Abalation Study

Differentiable Rendering Strategy. We conduct an ablation study to explore the effect of our differentiable rendering. As shown in Tab. III (Index I-II), rotation error decreased by

Category	Method	Per-part Pose			Joint State ↓
		Rotation Error (°) ↓	Translation Error (m) ↓	3D IOU (%) ↑	
Laptop	A-NCSH [21]	5.3, 5.4	0.054, **0.043**	56.7, 40.2	3.5°
	OMAD [23]	5.4, 4.3	0.062, 0.061	43.5, 24.1	3.3°
	ContactArt [24]	4.9, 4.7	**0.053**, 0.066	64.6, 50.4	5.8°
	CAPED	**3.6**, **3.8**	0.063, 0.069	**88.4**, **88.0**	**2.0°**
Eyeglasses	A-NCSH [21]	3.7, 22.3, 23.2	0.049, 0.313, 0.324	52.5, 40.2, 39.6	12.8°, 14.2°
	OMAD [23]	4.9, 7.5, 7.5	0.062, 0.103, 0.324	22.8, 20.5, 21.4	**4.9°**, 5.2°
	ContactArt [24]	4.1, **6.2**, 6.0	0.047, **0.095**, 0.091	58.6, 46.5, 51.7	5.6, 5.5°
	CAPED	**2.9**, 6.9, **5.7**	**0.039**, 0.097, **0.085**	**92.0**, **82.4**, **84.3**	5.0°, **4.1°**
Dishwasher	A-NCSH [21]	4.0, 4.8	0.059, 0.123	84.3, 56.2	3.8°
	OMAD [23]	6.0, 6.2	0.104, 0.142	66.5, 38.9	3.7°
	ContactArt [24]	3.9, 4.3	0.055, **0.079**	89.3, 67.6	6.0°
	CAPED	**2.6**, **2.9**	**0.050**, 0.083	**89.2**, **80.2**	**2.0°**
Scissors	A-NCSH [21]	**2,0**, 2.9	0.035, **0.025**	46.5, 44.8	4.4°
	OMAD [23]	3.9, 3.4	0.048, 0.039	35.6, 34.5	3.2°
	ContactArt [24]	2.2, **2.6**	**0.031**, 0.042	40.9, 46.3	4.2°
	CAPED	4.5, 4.6	0.045, 0.097	**74.3**, **68.9**	**2.5°**
Drawer	A-NCSH [21]	2.8, 3.5, 3.9, 2.9	0.045, 0.155, 0.157, **0.075**	90.2, 81.5, 78.4, 82.7	0.381m, 0.450m, 0.412m
	OMAD [23]	4.4, 4.4, 4.4, 4.4	0.111, 0.143, 0.144, 0.115	75.8, 73.4, 70.2, 71.3	0.110m, 0.111m, 0.092m
	ContactArt [24]	3.5, 3.5, 3.5, 3.5	0.061, 0.112, 0.121, 0.104	84.8, 78.6, 79.0, 81.2	0.076m, **0.074**m, 0.064m
	CAPED	**1.8**, **1.8**, **1.8**, **1.8**	**0.043**, **0.088**, **0.094**, 0.082	**91.2**, **85.3**, **85.3**, **86.5**	**0.075**m, 0.076m, **0.050**m

TABLE I: **Comparison with state-of-the-arts on the ArtImage dataset.** We validate our CAPED on categories Laptop, Eyeglasses, Dishwasher, Scissors and Drawer. ↓ means the lower the better and ↑ means the upper the better.

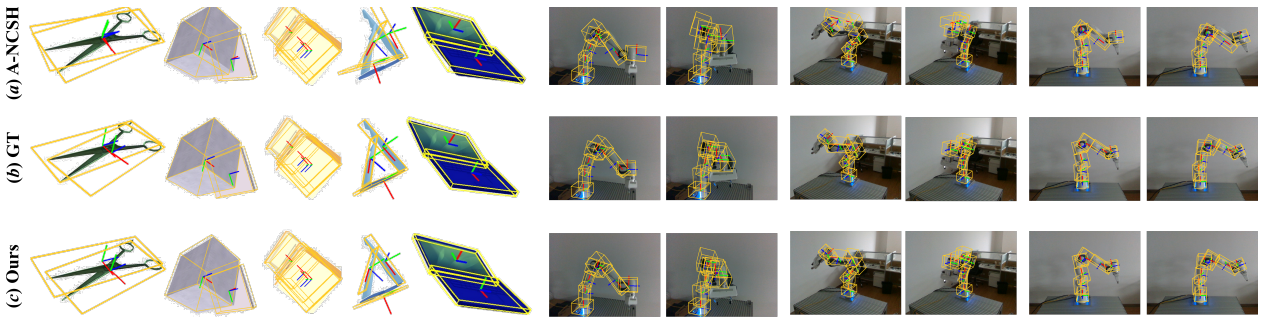

Fig. 3: **Qualitative results**. The left is the synthetic dataset (ArtImage), and the right is the real-world scenario (RobotArm).

Per-part Rotation Error (°)							
Part ID	1	2	3	4	5	6	7
A-NCSH [21]	7.8	7.9	10.3	10.5	11.2	16.4	23.5
CAPED	**0.08**	**6.2**	**7.3**	**7.6**	**8.2**	**12.3**	**16.7**
Per-part Translation Error (m)							
Part ID	1	2	3	4	5	6	7
A-NCSH [21]	**0.012**	0.044	0.067	0.066	**0.079**	0.236	0.403
CAPED	0.014	**0.024**	**0.048**	**0.052**	0.126	**0.128**	**0.312**

TABLE II: **Pose estimation results on RobotArm dataset.**

Index	Render	Rotation Error (°)	Translation Error (m)
I	-	1.8	0.092
II (Ours)	✓	**1.5**	**0.076**
Index	**Occlusion Level (Visibility)**	**Rotation Error (°)**	**Translation Error (m)**
III	0%-40%	1.5	0.057
IV	40%-80%	1.6	0.062
V	80%-100%	1.6	0.095

TABLE III: **Ablation study results**. The average Rotation (°) and Translation (m) of all parts are reported as the metrics. Note that experiments are conducted on the category *Drawer*.

16% and translation error decreased by 17%, proving that 2D Mask help the network acquire more effective features.

Self-Occlusion Analysis. To further investigate the robustness under self-occlusion of CAPED, we split the test samples of category *Drawer* into three subsets based on occlusion level. Thanks to our joint-centric method, rotation and translation errors are maintained stable with an increasing occlusion level. The results shown in Tab. III (Index III-V) show the effectiveness of CAPED in addressing the self-occlusion problem.

IV. Conclusion

In this paper, we introduce CAPED, a joint-centric approach for category-level articulation pose estimation that addresses self-occlusion via kinematic constraints. To refine pose predictions, we implement a differentiable binarization and rendering strategy. Experiments show our method achieves state-of-the-art performance on the synthetic ArtImage dataset and strong generalization on real-world datasets like RobotArm.

Acknowledgements

This research was supported by the National Natural Science Foundation of China (Grants 62373329, 62302143, 62372150), the Baima Lake Laboratory Joint Funds of the Zhejiang Provincial Natural Science Foundation (Grant LBMHD24F030002), the Zhejiang Provincial Special Support Program for High-Level Talents (2021R52004), and the Anhui Provincial Natural Science Foundation (Grants 2308085QF207, 2408085MF159). Partial support was also provided by Beijing XiaoYu Intelligence Manufacturing Ltd Inc, Beijing, China.

REFERENCES

[1] S. Y. Gadre, K. Ehsani, and S. Song, "Act the part: Learning interaction strategies for articulated object part discovery," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 15 752–15 761.

[2] L. Zhang, Z. Han, Y. Zhong, Q. Yu, X. Wu *et al.*, "Vocapter: Voting-based pose tracking for category-level articulated object via inter-frame priors," in *ACM Multimedia 2024*, 2024.

[3] L. Zhang, W. Meng, Y. Zhong, B. Kong, M. Xu, J. Du, X. Wang, R. Wang, and L. Liu, "U-cope: Taking a further step to universal 9d category-level object pose estimation," in *European Conference on Computer Vision*. Springer, 2025, pp. 254–270.

[4] H. Wang, S. Sridhar, J. Huang, J. Valentin, S. Song, and L. J. Guibas, "Normalized object coordinate space for category-level 6d object pose and size estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2642–2651.

[5] C. Wang, D. Xu, Y. Zhu, R. Martín-Martín, C. Lu, L. Fei-Fei, and S. Savarese, "Densefusion: 6d object pose estimation by iterative dense fusion," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 3343–3352.

[6] D. Wesierski and A. Jezierska, "Instrument detection and pose estimation with rigid part mixtures model in video-assisted surgeries," *Medical image analysis*, vol. 46, pp. 244–265, 2018.

[7] R. Brégier, F. Devernay, L. Leyrit, and J. L. Crowley, "Defining the pose of any 3d rigid object and an associated distance," *International Journal of Computer Vision*, vol. 126, no. 6, pp. 571–596, 2018.

[8] D. Amin and S. Govilkar, "Comparative study of augmented reality sdks," *International Journal on Computational Science & Applications*, vol. 5, no. 1, pp. 11–26, 2015.

[9] E. Marchand, H. Uchiyama, and F. Spindler, "Pose estimation for augmented reality: a hands-on survey," *IEEE transactions on visualization and computer graphics*, vol. 22, no. 12, pp. 2633–2651, 2015.

[10] Y. Su, J. Rambach, N. Minaskan, P. Lesur, A. Pagani, and D. Stricker, "Deep multi-state object pose estimation for augmented reality assembly," in *2019 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*. IEEE, 2019, pp. 222–227.

[11] N. Haouchine, P. Juvekar, M. Nercessian, W. M. Wells III, A. Golby, and S. Frisken, "Pose estimation and non-rigid registration for augmented reality during neurosurgery," *IEEE Transactions on Biomedical Engineering*, vol. 69, no. 4, pp. 1310–1317, 2021.

[12] L. Zhang, Y. Zhong, J. Wang, Z. Min, L. Liu *et al.*, "Rethinking 3d convolution in ℓ_p-norm space," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[13] Y. Chen, S. Huang, T. Yuan, S. Qi, Y. Zhu, and S.-C. Zhu, "Holistic++ scene understanding: Single-view 3d holistic scene parsing and human pose estimation with human-object interaction and physical commonsense," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 8648–8657.

[14] S. Huang, S. Qi, Y. Xiao, Y. Zhu, Y. N. Wu, and S.-C. Zhu, "Cooperative holistic scene understanding: Unifying 3d object, layout, and camera pose estimation," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[15] H. Xiong, H. Fu, J. Zhang, C. Bao, Q. Zhang, Y. Huang, W. Xu, A. Garg, and C. Lu, "Robotube: Learning household manipulation from human videos with simulated twin environments," in *Conference on Robot Learning*. PMLR, 2023, pp. 1–10.

[16] S. Alatartsev, S. Stellmacher, and F. Ortmeier, "Robotic task sequencing problem: A survey," *Journal of intelligent & robotic systems*, vol. 80, pp. 279–298, 2015.

[17] A. Collet, D. Berenson, S. S. Srinivasa, and D. Ferguson, "Object recognition and full pose registration from a single image for robotic manipulation," in *2009 IEEE International Conference on Robotics and Automation*. IEEE, 2009, pp. 48–55.

[18] B. An, Y. Geng, K. Chen, X. Li, Q. Dou, and H. Dong, "Rgbmanip: Monocular image-based robotic manipulation through active object pose estimation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 7748–7755.

[19] C. Pan, B. Okorn, H. Zhang, B. Eisner, and D. Held, "Tax-pose: Task-specific cross-pose estimation for robot manipulation," in *Conference on Robot Learning*. PMLR, 2023, pp. 1783–1792.

[20] S. Tyree, J. Tremblay, T. To, J. Cheng, T. Mosier, J. Smith, and S. Birchfield, "6-dof pose estimation of household objects for robotic manipulation: An accessible dataset and benchmark," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 13 081–13 088.

[21] X. Li, H. Wang, L. Yi, L. J. Guibas, A. L. Abbott, and S. Song, "Category-level articulated object pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3706–3715.

[22] L. Liu, W. Xu, H. Fu, S. Qian, Q. Yu, Y. Han, and C. Lu, "Akb-48: a real-world articulated object knowledge base," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 14 809–14 818.

[23] H. Xue, L. Liu, W. Xu, H. Fu, and C. Lu, "Omad: Object model with articulated deformations for pose estimation and retrieval," *arXiv preprint arXiv:2112.07334*, 2021.

[24] Z. Zhu, J. Wang, Y. Qin, D. Sun, V. Jampani, and X. Wang, "Contactart: Learning 3d interaction priors for category-level articulated object and hand poses estimation," in *2024 International Conference on 3D Vision (3DV)*. IEEE, 2024, pp. 201–212.

[25] L. Zheng, C. Wang, Y. Sun, E. Dasgupta, H. Chen, A. Leonardis, W. Zhang, and H. J. Chang, "Hs-pose: Hybrid scope feature extraction for category-level object pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 17 163–17 173.

[26] Y. Di, R. Zhang, Z. Lou, F. Manhardt, X. Ji, N. Navab, and F. Tombari, "Gpv-pose: Category-level object pose estimation via geometry-guided point-wise voting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 6781–6791.

[27] L. Liu, A. Huang, Q. Wu, D. Guo, X. Yang, and M. Wang, "Kpa-tracker: Towards robust and real-time category-level articulated object 6d pose tracking," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 4, 2024, pp. 3684–3692.

[28] O. Wiles, G. Gkioxari, R. Szeliski, and J. Johnson, "Synsin: End-to-end view synthesis from a single image," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 7467–7477.

[29] R. Gong, X. Liu, S. Jiang, T. Li, P. Hu, J. Lin, F. Yu, and J. Yan, "Differentiable soft quantization: Bridging full-precision and low-bit neural networks," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 4852–4861.

[30] L. Liu, H. Xue, W. Xu, H. Fu, and C. Lu, "Toward real-world category-level articulation pose estimation," *IEEE Transactions on Image Processing*, vol. 31, pp. 1072–1083, 2022.