DFGAP: Towards Depth-Free Cross-Category GAParts Perception via Uncertainty-Quantified Modeling

Xueyu Yuan*
Hefei University of Technology
Hefei, China
yxyxc0410@mail.hfut.edu.cn

Jiarui Zhang*
Hefei University of Technology
Hefei, China
jiaruizhang@mail.hfut.edu.cn

Jiangqi Song
Hefei University of Technology
Hefei, China
jiangqisong@mail.hfut.edu.cn

Liu Liu†
Hefei University of Technology
Hefei, China
liuliu@hfut.edu.cn

Li Zhang
University of Science and Technology of China
Hefei, China
zanly20@mail.ustc.edu.cn

Dan Guo
Hefei University of Technology
Hefei, China
guodan@hfut.edu.cn

Richang Hong
Hefei University of Technology
Hefei, China
hongrc@hfut.edu.cn

Meng Wang
Hefei University of Technology
Hefei, China
wangmeng@hfut.edu.cn

Abstract

Cross-category object perception is one of the essential upstream tasks for generelizable robot object interaction and manipulation. Recently, an increasing number of researchers are focusing on investigating visual Generalizable and Actionable Parts understanding at cross-category level perception. However, these works are built upon the RGB-D or point cloud input, that relies on the depth information capture. Under the circumstances of limited depth camera performance, e.g. transparent or light absorbing material, perception algorithms that do not require depth information are urgently needed. In this paper, we propose DFGAP, a novel depth-free framework for RGB-based GAParts segmentation and pose estimation. Specifically, we independently model the ill-pose problems from the absence of depth for GAPart segmentation and pose estimation, by clearly quantifying the pixel-wise segmentation probability and relative depth. We reduce the uncertainty and benefit learning in these two tasks. The experimental results demonstrate the superior performance and robustness of our DFGAP. Our work provides a new research paradigm in GAParts perception. We believe that our work has the enormous potential to be applied in many areas of embodied AI system.

CCS Concepts

• **Computing methodologies** → **Vision for robotics**.

*Both Xueyu Yuan and Jiarui Zhang contributed equally to this research.
†Liu Liu is the corresponding author.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
MM '25, Dublin, Ireland
© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-2035-2/2025/10
https://doi.org/10.1145/3746027.3755113

Keywords

Embodied Intelligence, Generalizable and Actionable Part, Depth-Free Object Perception

ACM Reference Format:
Xueyu Yuan, Jiarui Zhang, Jiangqi Song, Liu Liu, Li Zhang, Dan Guo, Richang Hong, and Meng Wang. 2025. DFGAP: Towards Depth-Free Cross-Category GAParts Perception via Uncertainty-Quantified Modeling . In *Proceedings of the 33rd ACM International Conference on Multimedia (MM '25), October 27–31, 2025, Dublin, Ireland*. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3746027.3755113

1 Introduction

Generalizable object perception and manipulation is a fundamental task for developing various embodied artificial intelligence systems [1, 8, 24, 37]. Different from instance-level or category-level object analysis, generalizable object perception requires the agent to understand cross-category parts and their corresponding operational functions from an unseen object category. In this context, Geng et al. [6] introduce the concept of Generalizable and Actionable Parts (GAParts) which enables the model to focus on the common parts of objects that exhibit similar affordance, which provides an innovative paradigm for generalizable object manipulation investigation.

To solve the problem of GAParts segmentation and 6D pose estimation task, existing approaches take RGB-D image or point cloud as visual input and build a corresponding 3D vision processing network [6, 13]. This solution may face several challenges: (1) High-precision depth sensor is required when transferring these methods from the simulation to the real world, while the current depth camera might not satisfy this demand. (2) For these methods, taking a single RGB image as input may bring two key issues: shape variation and scale ambiguity. These challenges arise from the uncertainty caused by the lack of depth data, making the estimation of both rotation and translation an ill-posed problem.

In this paper, we aim to eliminate the limitation of relying on depth information, and propose **DFGAP**, a novel framework for

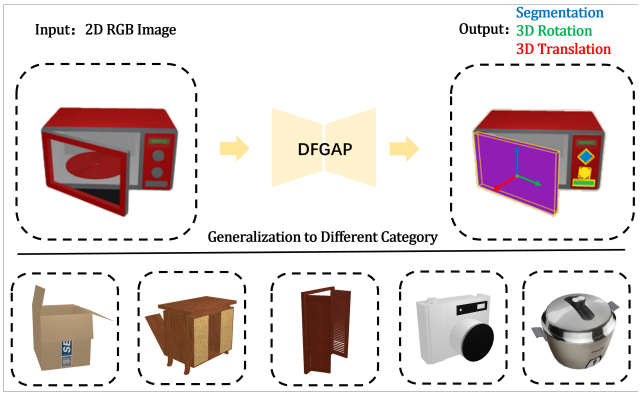

Figure 1: GAParts Perception Task. Given a single RGB image, we aim to acquire GAParts segmentation and per-GAPart pose estimation, further generalize to other objects of various category.

Depth-**F**ree cross-category **GA**Part **P**erception. Due to the uncertainty problem in modeling object shape and scale, our method independently quantifies the uncertainties in both GAPart segmentation and pose estimation stages, which enables depth-free GAParts perception to realize superior performance compared to those using depth information. Specifically, in the segmentation stage, we use RollingUNet [23] as the backbone to extract global features. The main challenge lies in the uncertainty of assigning each pixel due to the lack of depth information. To address this, we introduce a mixture probability model, which robustly and accurately learns the segmentation of GAParts. This model quantifies the probability that each pixel belongs to different components, preventing unreasonable segmentation distributions. We employ a mixture probability modeling branch to predict the distribution map based on a Gaussian distribution. Using the global features and the mixture distribution map as input, we then apply a Mask-RCNN[9] network to obtain the final GAPart segmentation.

In terms of pose estimation stage, current methods aim to reduce ambiguity caused by the absence of depth and introduce additional input or performing normalization to target object to resolve this issue. But this strategy still faces the problems of shape variation and scale ambiguity. Thus, we suggest directly estimating the relative distance to a predetermined depth by explicitly quantifying the uncertainty of each GAPart's depth with minimal external priors. In detail, our DFGAP first extracts global features from the segmented GAParts using the pre-trained DINOv2[25] as the backbone. Rather than directly predicting the depth of each pixel, we propose predicting a set of keypoints using a self-supervised method. Subsequently, we estimate the depth of these predicted keypoints through another self-supervised network. To enhance this process, we introduce a spherical harmonic-based 3D encoding scheme to convert the predicted depth information into high-dimensional features. Finally, using both the encoded depth features and the global features, we employ three parallel branches to independently predict the x-axis rotation, y-axis rotation, and translation vector, thereby recovering the complete 6D pose.

We evaluate our DFGAP in GAPartNet dataset, which is derived from two well-known benchmarks PartNet-Mobility[32] and AKB-48[38]. Extensive experiments show the superior performance of DFGAP in GAPart segmentation and pose estimation tasks, compared to RGB-D methods and depth-free methods. Based on the high-quality GAPart perception results, we build a simple heuristic object interaction policy for a robot arm with gripper. The demonstrations illustrate with help of our DFGAP, the robot agent can achieve generalizable object manipulation task in both the simulation and the real world under a single RGB image input. We believe that our work has the enormous potential to be applied in many areas of embodied AI system.

In summary, our contributions can be concluded as follows:

- We propose a novel framework DFGAP, that first solves depth-free GAPart perception task, enabling cross-category object perception and eliminating the need for additional depth sensors.
- Our DFGAP achieves uncertainty-quantified modeling in both GAParts segmentation and pose estimation stage. For segmentation stage, we propose a mixture probability modeling that quantifies the probability of each pixel labels. For pose estimation stage, we also design a method to mitigate the uncertainty by using self-supervised keypoints learning.
- DFGAP shows superior performance in GAPart segmentation and pose estimation tasks compared to those containing depth information. The perception results also help the robot manipulate the object not only in the simulation but also in the real world.

2 Related Works

2.1 Cross-Category Object Perception

As a downstream task of Embodied intelligence, object perception has been studied at both the instance level [26–28, 33] and the category level [3, 4, 12, 17, 30, 38]. Although some previous category-level methods [1, 7, 24] have certain generalizability on unseen categories, cross-category object perception remains challenging and has not been fully explored. [37] proposes GenPose, a novel solution that reframes category-level object pose estimation as a conditional generative modeling task, and demonstrates consistent and robust performance on symmetrical unseen categories. Xu *et al.* propose CAPE [34], which aims to create a 2D pose estimation model capable of detecting the pose of any class of object given only a few samples with keypoint definition. To handle novel object categories, especially articulated objects and movable parts, Geng *et al.* propose to learn such cross-category skills via Generalizable and Actionable Parts(GAParts) [6], which are generalizable in both recognition and manipulation. PartSLIP[21] extends 3D part segmentation leveraging pre-trained image-language models. [16] proposes a framework named Affinity3D that intends to empower semantic segmentation models to perceive novel samples by combining the geometric separation in 3D and the zero-shot capabilities of 2D models.

2.2 Depth-free Based Pose Estimation

In recent years, RGBD-based methods for category-level object pose estimation have been fully explored [10, 14, 18, 30, 35, 39]. As for depth-free methods, recent work has focused on dealing with

the absence of depth, which presents scale ambiguity and shape variation. Xu *et al.* combine a gradient-based fitting procedure with a parametric neural image synthesis module, which could implicitly represent the appearance, shape and pose of entire object categories for the pure RGB case [2]. Fan *et al.* innovate OLD-Net, a depth-free model that predicts the pose via estimating Normalized Object Coordinate Space (NOCS) coordinates, while incorporating global position hints and shape priors [5]. Although this method establishes 3D-3D correspondences by coordinates prediction and metric depth estimation, scale ambiguity is not accounted for in this method. Another method is introduced by [31], which proposes a novel pipeline that decouples the 6D pose and size estimation. Inlier 2D-3D correspondence and metric scale recovery are established by using a pre-trained monocular estimator. Zhang *et al.* propose Lapose [40], a novel framework that models the shape of the object as the Laplacian mixture model for pose estimation. By representing each point as a probabilistic distribution, the shape uncertainty could be explicitly quantified.

3 Problem Formulation

Following the definition of GAPart in GAPartNet, the overview of our framework and problem formulation is as follows:

Given a single RGB image containing an object of a particular category $I \in \mathbb{R}^{H\times W\times 3}$, where H and W denote the height and width of the input image. We assume that the object in the image contains L GAParts, with part labels ranging from $\{1,\ldots,L\}$.Our GAParts segmentation framework aims to acquire the final per-GAParts segmentation S^k. $S^k \in \{1,\ldots,L\}$ denotes the pixel-wise instance segmentation label. With these segmented GAPart, our pose estimation framework is a response to precisely predicting the 6 DoF pose T_k for the k-th GAPart, including 3 DoF rotation R_k and 3 DoF translation t_k, where k ranges from $\{1,\ldots,L\}$.

4 Methods

4.1 GAParts Segmentation

4.1.1 ***Segmentation Motivation and Overview.*** With the input image I, the aim of our segmentation network is to obtain the final pixel-wise GAParts segmentation S_k. Several previous works have demonstrated the applications of probability modeling in multiple perception and vision tasks. As for segmentation, distributing the pixel-wise segmentation label, especially for those closed to different GAParts, is a highly uncertain question which is suitable for solving through probability modeling. Additionally, we notice that, normally, the final segmentation results are somewhat unreasonable and unbalanced in terms of spatial distribution. Therefore, we introduce this method to reduce the uncertainty in the segmentation results as well as to make the GAParts segmentation results more reasonably distributed. Firstly, we leverage a backbone to extract global pixel-wise features $\mathcal{F}_{global}$. Then by modeling a probability distribution map, we obtain $\mathcal{F}_{prob}$. Finally, we concatenate the dual stream output $\mathcal{F}_{global}$ and $\mathcal{F}_{prob}$, as input to a Mask RCNN network to predict the final GAParts segmentation S_k.

4.1.2 ***Mixture Probability Modeling.*** Given a pixel-wise global feature extracted from the backbone, we first attach a simple network to encode normalized coordinate information in the feature map. Then, to model the mixture probability distribution map, following some previous works, we design a lightweight network to estimate a set of gaussian mixture parameters (μ_k, σ_k,ρ_k) for k parts. Moreover, differing from previous approaches, we also estimate the mixture weight π_k and the background probability P_{bg}, this is motivated by the following two considerations. On the one hand, we notice that a simple estimation of distribution center μ_k and distribution scale σ_k^2 may lead to uncertainty in final probability map modeling , causing a high probability that the probability map result will collapse globally as the background. On the other hand, since most areas of the input RGB image consist of meaningless background, separately modeling the pixel-wise probability distribution of the background helps to better distinguish objects from the background, thereby reducing uncertainty when modeling object-associated regions. We attach an independent branch for the estimation of each parameter. To ensure rationality, we make some minor adjustments to estimated parameters. Specifically, we apply an exponential transformation to the parameter μ_k, in order to ensure it is positive, and apply a sigmoid function to the parameter σ_k^2 in order to restrict its range to $(0, 1)$.

We generate the pixel-wise mixture probability distribution with predicted gaussian parameters. More detailed, firstly, for each pixel i in the k-th GAPart, we calculate its normalized offset relative to gaussian distribution center $\mu_{k,x}$ and $\mu_{k,y}$ as in:

$$\mathrm{d}_{x,i} = \frac{x_i - \mu_{x,k}}{\sigma_{x,k}}, \quad \mathrm{d}_{y,i} = \frac{y_i - \mu_{y,k}}{\sigma_{y,k}} \tag{1}$$

Where $\mathrm{d}_{x,i}$ and $\mathrm{d}_{y,i}$ denote normalized offset in x and y directions, and $(x_i,\ y_i)$ denotes the 2d coordinate of pixel i in image I. Then, we generate probability of each pixel by calculating 2D gaussian density function, which can be defined as:

$$z_i = \frac{d_{x,i}^2 + d_{y,i}^2 - 2\rho_k d_{x,i} d_{y,i}}{2\left(1-\rho_k^2\right)} \tag{2}$$

$$p\left(x_i, y_i\right) = \frac{e^{-z_i}}{2\pi\sigma_{x,k}\sigma_{y,k}\sqrt{1-\rho_k^2}} \tag{3}$$

Finally, we use the predicted mixture weight π_k to modify the $p\left(x_i, y_i\right)$ to obtain the final probability as follows:

$$p_{i,k} = \pi_k p\left(x_i, y_i\right) \tag{4}$$

Within the pixel-wise probability $p_{i,k}$, we can generate a probability distribution P_k of $k-th$ GAPart. In addition, by concatenating the k gaussian probability distribution maps and the background probability map P_{bg} we have obtained, we achieve the final gaussian mixture probability distribution map P. Finally, we attach a convolutional layer to adjust P to probability features $\mathcal{F}_{global}$.

4.1.3 ***Loss Function.*** We design a set of combined loss functions to better supervise the learning of gaussian mixture probability distribution. Firstly, we follow the previous work to employ standard gaussian negative log-likelihood loss (NLL) to supervise the predicted parameters (μ_k, σ_k) for the $k-th$ GAPart as in:

$$\mathcal{L}_{param} = \sum_{k=1}^{L} \frac{\lambda_k \left\| \mu_k - c_{gt}^k \right\|^2}{2\sigma_k^2} + log(\sigma_k) \tag{5}$$

Please note that c_{gt}^k here means the ground truth geometry center of the $k-th$ GAPart, λ_k is pre-defined hyper-parameter.

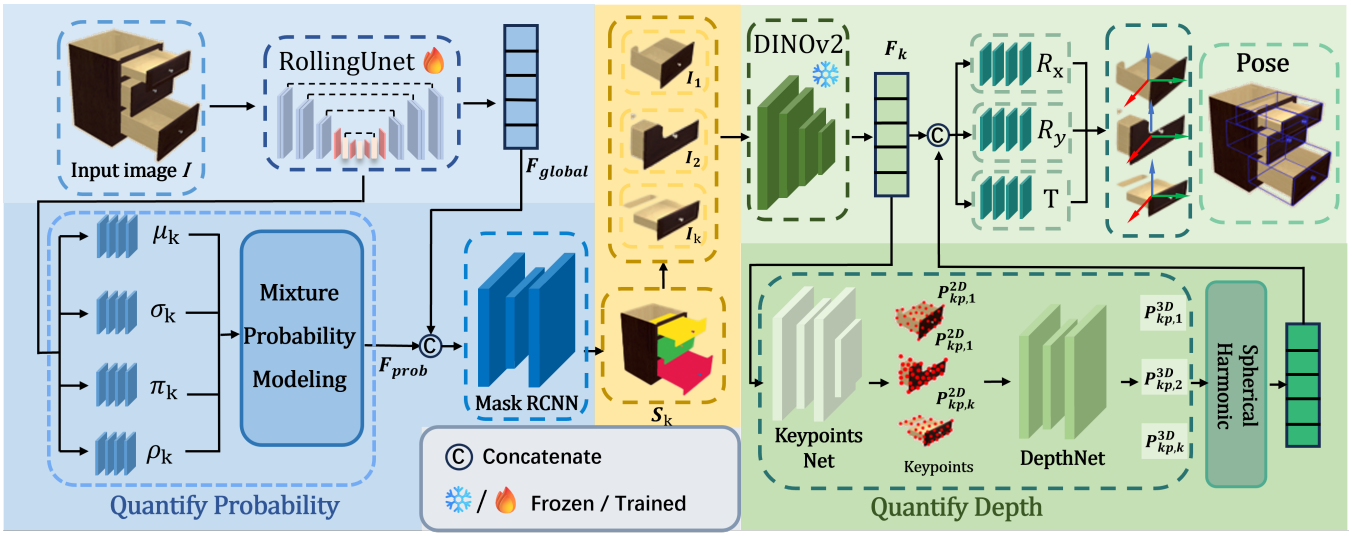

Figure 2: An Overview of Our framework We firstly predict per-GAPart segmentation by modeling the mixture probability distribution.(blue section). Then, we design three key components to estimate the per-GAPart 6 Dof pose. (green section)

Then, we encourage the predicted π_k close to the ground truth distribution of the $k-th$ GAPart. To achieve this target, we first obtain the normalized distribution ratio as ground truth π_{gt}^k of the $k-th$ GAPart by calculating the number of pixels on the input image I. In addition, we spatially average the predicted π_k . Finally, we take Kullback-Leibler divergence (KL divergence) to force the spatially consistent of π_{gt}^k and π_k. Therefore, the global mixture weight is supervised by following loss item:

$$\mathcal{L}_\pi = \sum_{k=1}^{L} KL(softmax(\frac{\pi_k}{T_k}),\ \pi_{gt}^k) \tag{6}$$

Where KL denotes the KL divergence, T_k is a hyper-parameter.

To refine the probability distribution, especially for slight GAParts, we also apply pixel-wise supervision between predicted probability distribution. Here we take focal loss[15] since its outstanding performance in imbalanced categories. The loss item is as follows:

$$\mathcal{L}_{\text{FL}} = -\alpha(1-S_k)^\gamma log(S_k) \tag{7}$$

In summary, the final GAParts segmentation loss is a linear weighted combination of all loss items $\mathcal{L}_{param}$, $\mathcal{L}_\pi$ and $\mathcal{L}_{\text{FL}}$

4.2 GAParts Pose Estimation

4.2.1 ***Pose Estimation Motivation and Overview.*** Given the predicted segmentation label S_k, we extract the image I_k which contains only the $k-th$ GAPart. we aim to obtain the 6D pose, including rotation R_k and T_k by our pose estimation framework. Most depth-free pose estimation methods focus on reducing the ambiguity caused by the absence of depth data. Typically,by introducing additional input or performing normalization to target object to resolve this issue. However, it does not solve two common issues, shape variation and scale ambiguity, in depth-free pose estimation essentially, but it introduces additional uncertainty. Therefore, we suggest returning to the essence of the problem and propose to directly estimate the relative distance to a predetermined depth. To begin with, we leverage DINOv2[25] as the backbone to extract global features. Then, we attach two self-supervised networks to predict a group of keypoints and their corresponding depth. After that, we designed a spherical harmonic based encoding scheme to acquire the high-dimension features. Finally, our pose estimation network predicts the 6D pose (R_k, T_k) for the $k-th$ GAPart.

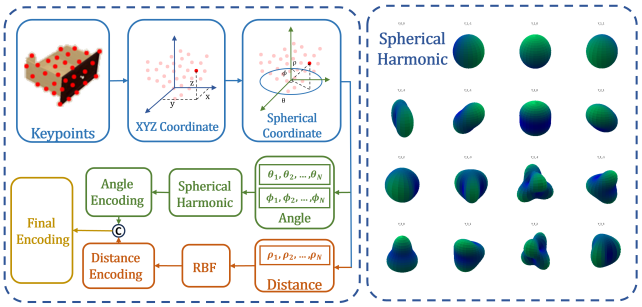

Figure 3: Spherical Harmonic Encoding . First: We transform the keypoints from a XYZ system to a spherical system; Second: We take a set of spherical harmonics and radial basis functions to encode direction and distance respectively.

4.2.2 ***Self-Supervised keypoints prediction*** . As we have discussed above, we first attach a self-supervised network to predict a group of keypoints. We establish two principles. First, keypoints should cover the areas in the image where the pixel gradient changes dramatically, as these areas are usually where the shape of the object changes. Second, keypoints should be evenly distributed on the surface of the object to ensure the keypoints are able to represent the object globally.

Our keypoints network take global feature $\mathcal{F}_k$, which is extracted by DINOv2, as input, directly predict the normalized coordinate of 2D keypoints P_{kp}^{2D}. To achieve these objectives, we design the following two loss items to promote the learning.

We calculate pixel-wise gradient for the input image I_k and normalize to (0,1) to obtain a gradient map G_k for the $k-th$ GAPart. For each predicted keypoint $p_i^{2D} \in P_{kp}^{2D}, i \in [1, N]$, where N denotes the number of predicted keypoints. We minimize the following item:

$$\mathcal{L}_{grad} = \frac{1}{N}\sum_{i=1}^{N}\left[1 - G\left(p_i^{2D}\right)\right] \tag{8}$$

To force the even covering, we calculate the normalized distance between each pair of keypoints $\left(p_i^{2D}, p_j^{2D}\right),\ i, j \in [1, N]$, and then supervise the even distribution of keypoints through the following loss item:

$$\mathcal{L}_{dist} = \frac{1}{N(N-1)}\sum_{i=1}^{N}\sum_{j=1, j\neq i}^{N}\left[max\left(0, \delta - \left\|p_i^{2D} - p_j^{2D}\right\|_2\right)\right]^2 \tag{9}$$

Please note that δ is a hyper-parameter, which means the minimum distance allowed between two keypoints (p_i^{2D}, p_j^{2D}). Finally, our self-supervised keypoints loss function is a linear combination of the two loss items mentioned above.

4.2.3 ***Self-Supervised keypoints depth estimation*** . Given predicted keypoints P_{kp}^{2D}, we then attach a depth network to estimate the point-wise relative distance to a predetermined depth. Specifically, we firstly extract corresponding keypoints feature $\mathcal{F}_{kp}$ from global feature $\mathcal{F}_k$. then, taking $\mathcal{F}_{kp}$ as input, our depth network estimate the relative distance Z_{kp} as output. With the P_{kp}^{2D} and Z_{kp}, we can easily recover the 3D keypoint p_i^{3D}. To make the predicted keypoint depth more reliable, we design three loss functions based on geometric constraints.

When 2D keypoints are projected into 3D space through camera intrinsic, if the predicted depth Z_{kp} are precise, the geometry structure of keypoints should maintain relative stability. Specifically, we define d_{ij}^{3D} as the distance between keypoints p_i^{3D} and p_j^{3D}, d_{ik}^{2D} as the distance between keypoints p_i^{2D} and p_j^{2D}. The 3D distance should be proportional to its 2D distance on the normalized camera plane. In other words, the ratio between two distances , we define as:

$$r_{ij} = \frac{d_{ij}^{3D}}{d_{ik}^{2D} + \epsilon} \tag{10}$$

should be approximately equal to the average depth $\bar{z}$ of the keypoints, which defined as:

$$\bar{z} = \frac{1}{N}\sum_{i=1}^{N} z_i \tag{11}$$

Please note that in order to avoid dividing by zero, we add a very small constant ϵ to the denominator, normally 1e-6.

Relying on this geometric constraint, we attach the geometry-limitation loss as follows to supervise the learning of keypoints' depth.

$$\mathcal{L}_{geo} = \frac{2}{N(N-1)}\sum_{i<j}\left|r_{ij} - \bar{z}\right| \tag{12}$$

All symbols here have the same meaning with those we have mentioned above.

Additionally, based on a universal consensus that the depth of keypoints that are closer should be close. We design the depth smoothness loss item as follow:

$$\mathcal{L}_{smooth} = \frac{2}{N(N-1)}\sum_{i<j}\omega_{ij}\left|\ell_i - \ell_j\right| \tag{13}$$

Here we take $\ell_i = log(z_i + \delta)$ to perform logarithmic smoothing to the depth z_i. And ω_{ij} is used to control the loss weight of point pairs (p_i, p_j)

However, another obvious fact is that keypoints located at the edge of object, even if they are close in distance, will experience drastic changes in depth. Based on that, we must take into account regions with significant gradient changes in the image, since they normally representative the edge of the object. Therefore, we firstly compute the gradient in x-axis $\nabla_{x,i}$ and in y-axis $\nabla_{y,i}$ for each keypoint p_i^{2D}, then minimize the following loss items:

$$\mathcal{L}_{edge} = \frac{1}{N}\sum_{i=1}^{N}\omega_i\left(\left|\nabla_{x,i} z_i\right| + \left|\nabla_{y,i} z_i\right|\right) \tag{14}$$

Also , ω_i here is a edge weight, and α is a hyper-parameter that regulates the rate of edge weight decay. In summary, the final loss function of our self-supervised depth estimation is the combination of all loss item mentioned above.

4.2.4 ***3D Encoding Based on Spherical Harmonic Function***. As we mentioned above, obtaining the predicted depth Z_{kp}, we can easily recover the 3D coordinate then obtain the 3D keypoints P_{kp}^{3D}. In order to enable the network to utilize the synthesized 3D position information more efficiently, a common approach is designing an encoding scheme to encode the position into high-dimension features. Previous work's achievements inspire us to consider the pose learning,especially for rotation learning, in spherical coordinate space rather than the traditional xyz coordinate space. Since the spherical coordinate space represents direction and distance more intuitively using two angles (φ, θ) and a radial distance ρ, we propose using the spherical harmonic function to encode 3D spherical coordinates into high-dimensional features. This choice is due to its excellent properties in rotation prediction tasks. Mathematically speaking, spherical harmonics possess strict orthogonality, completeness, and rotational equivariance, providing a set of continuous, smooth, and rotation-equivariant basis functions that can encode an object's rotation information into high-dimensional features. Even when rotation angles approach the boundaries (for instance, near 0 or π), the encoded output remains varying smoothly. This characteristic effectively mitigates the training difficulties caused by the discontinuities inherent in the topology of SO(3) (such as singularities and double-cover issues), thereby facilitating robust learning of rotation. The process is shown as Fig.3. Follow the above instruction, given a 3D keypoint $p_i^{3D} = (x_i, y_i, z_i) \in P_{kp}^{3D}$, we first transform the coordinate into a spherical coordinate space by following the formula:

$$\begin{cases} \rho = \sqrt{x_i^2 + y_i^2 + z_i^2} \\ \theta = \arccos\left(\frac{z}{\rho}\right) \\ \phi = \arctan\left(\frac{y}{x}\right) \end{cases} \tag{15}$$

We use Gaussian radial basis functions (rbf) to encode distances ρ. Specifically, firstly, we employ a set of N_{rbf} Gaussian RBFs and define a set of centers $\{c_i\}_{i=1}^{N_{rbf}}$ uniformly spaced in a pre-defined interval (e.g. [0,10]). For each center c_i, we compute the RBF response as:

Table 1: GAParts Segmentation Result

	Method	Ln.F.HI	Rd.F.HI	Hg.HI	Hg.Ld	Sd.Ld	Sd.Bn	Sd.Dw	Hg.Dr	Hg.Kb	Avg.AP	Avg.AP50
Seen	PointGroup[11]	86.1	23.0	84.6	80.0	88.3	49.3	62.6	92.8	34.6	57.3	66.8
	SoftGroup[29]	57.8	**93.6**	81.2	76.0	89.3	25.2	50.8	93.9	51.5	58.5	68.8
	AutoGPart[22]	86.8	20.3	87.7	79.7	89.4	62.3	61.6	92.5	16.7	57.2	66.3
	GAPartNet[6]	89.2	54.9	90.4	84.8	89.8	66.7	67.2	94.7	52.9	67.6	76.5
	CSS-GASP[36]	90.5	57.5	90.3	81.6	90.8	54.5	66.1	**94.8**	63.1	65.5	76.6
	GASEM[20]	65.1	69.4	58.6	62.9	66.1	17.2	75.3	67.9	58.7	52.3	59.6
	DFGAP	**90.8**	65.7	**92.1**	**85.9**	**95.2**	**68.7**	**95.5**	91.5	**70.1**	**74.8**	**83.2**
Unseen	PointGroup[11]	32.4	9.8	2.1	26.8	0.0	42.6	57.0	63.9	1.7	21.9	26.3
	SoftGroup[29]	25.8	5.0	0.4	33.9	0.6	50.9	51.2	69.0	12.1	22.0	27.7
	AutoGPart[22]	45.6	4.8	3.1	34.3	0.0	47.8	64.1	63.1	11.5	25.7	30.5
	GAPartNet[6]	45.6	40.0	3.1	40.2	5.0	49.1	64.2	69.1	23.4	32.0	37.2
	CSS-GASP[36]	48.2	42.1	1.1	**57.4**	0.1	41.9	65.7	69.3	31.5	31.1	38.9
	GASEM[20]	30.9	26.0	**3.4**	41.7	3.4	25.6	46.5	53.3	11.7	28.9	34.2
	DFGAP	**50.7**	**47.2**	2.3	53.8	**7.3**	**51.5**	**69.1**	**72.6**	**31.8**	**36.2**	**42.8**

Results of Part Segmentation in terms of Per-GAPart-class AP50 (%), Average AP50 (%), and Average AP (%). Ln.=Line. F.=Fixed. Rd.=Round. Hg.=Hinge. Hl.=Handle. Sd.=Slider. Ld.=Lid. Bn.=Button. Dw.=Drawer. Dr.=Door. Kb.=Knob.

Table 2: GAParts Pose Estimation Result

		Method	Modalities	Ln.F.HI	Rd.F.HI	Hg.HI	Hg.Ld	Sd.Ld	Sd.Bn	Sd.Dw	Hg.Dr	Hg.Kb	Avg
Seen	**Rotation(°)**	GAPartNet[6]	RGB-D	10.39	12.74	10.15	7.72	7.82	2.41	5.03	8.46	4.74	7.72
		CSS-GASP[36]	RGB-D	**5.33**	5.46	7.61	6.51	6.52	4.67	1.51	6.31	**4.22**	5.35
		GASEM[20]	RGB-D	17.11	11.96	9.42	6.93	4.45	9.11	9.30	8.44	5.31	9.11
		LaPose[40]	RGB	13.69	8.82	9.38	15.12	5.97	10.65	6.14	14.98	7.73	10.28
		R^2-Art[13]	RGB	13.76	18.57	9.24	11.89	8.82	14.63	7.79	26.38	14.13	13.91
		DFGAP	RGB	7.25	**0.71**	**4.48**	**6.28**	**4.21**	**0.78**	**1.41**	**5.67**	10.65	**4.60**
	Translation(cm)	GAPartNet[6]	RGB-D	0.015	0.078	0.039	0.038	0.032	0.009	0.075	0.038	0.010	0.037
		CSS-GASP[36]	RGB-D	0.012	0.076	0.037	0.022	0.021	**0.005**	0.042	0.025	0.011	0.028
		GASEM[20]	RGB-D	0.015	0.051	0.031	0.080	0.023	**0.005**	0.069	0.047	0.005	0.036
		LaPose[40]	RGB	0.018	0.107	0.046	0.053	0.028	0.011	0.042	0.028	0.023	0.040
		R^2-Art[13]	RGB	0.024	0.127	0.054	0.063	0.041	0.019	0.079	0.066	0.038	0.057
		DFGAP	RGB	**0.011**	**0.036**	**0.028**	**0.021**	**0.018**	0.007	**0.034**	**0.024**	**0.004**	**0.020**
Unseen	**Rotation(°)**	GAPartNet[6]	RGB-D	36.54	19.89	64.31	19.18	29.17	9.21	14.62	38.57	16.89	27.60
		CSS-GASP[36]	RGB-D	23.85	12.79	21.23	14.34	17.07	**6.51**	3.57	18.09	18.37	15.09
		GASEM[20]	RGB-D	33.85	27.41	57.66	19.63	24.04	8.28	20.72	22.62	**12.86**	25.23
		LaPose[40]	RGB	25.74	15.67	24.87	19.95	15.13	12.44	10.18	28.79	17.68	18.94
		R^2-Art[13]	RGB	47.38	37.88	45.21	26.87	32.19	18.28	22.37	40.05	14.55	31.64
		DFGAP	RGB	**18.74**	**2.14**	**17.99**	**12.76**	**10.75**	7.64	**2.87**	**16.95**	14.42	**11.58**
	Translation(cm)	GAPartNet[6]	RGB-D	0.164	0.091	0.539	0.415	0.076	0.042	0.318	0.131	0.038	0.202
		CSS-GASP[36]	RGB-D	0.138	0.078	0.242	0.345	**0.011**	0.026	0.027	0.045	0.026	0.104
		GASEM[20]	RGB-D	0.226	0.052	0.261	0.294	0.385	0.014	0.165	0.087	0.019	0.167
		LaPose[40]	RGB	0.278	0.107	0.296	**0.165**	0.283	0.022	0.219	0.067	0.018	0.162
		R^2-Art[13]	RGB	0.341	0.127	0.237	0.228	0.459	0.036	0.522	0.203	0.083	0.248
		DFGAP	RGB	**0.109**	**0.063**	**0.187**	0.197	0.046	**0.011**	**0.023**	**0.037**	**0.016**	**0.077**

Results of GAParts Pose estimation in terms of per-part-class Rotation error and translation error.. We use degree error (noted as °) and distance error (noted as cm) as metrics.

$$\phi_i(\rho) = exp\left(-\frac{(\rho - c_i)^2}{2\sigma^2}\right) \tag{16}$$

Where σ is a pre-determined fixed standard deviation (e.g. 1.0). Then, we apply a learned linear transformation $W_\rho \in \mathbb{R}^{d_\rho \times N_{rbf}}$ as:

$$\mathcal{F}_\rho = W_\rho \cdot \begin{bmatrix} \phi_1(\rho) \\ \phi_2(\rho) \\ \cdots \\ \phi_{N_{rbf}}(\rho) \end{bmatrix} \tag{17}$$

resulting in a d_ρ-dimensional encoding feature.

For two angle θ and ϕ, we take spherical harmonics to jointly encode the angular information instead of independently encoding each angle. Given a pair of angle (θ, ϕ). We firstly define real spherical harmonics function as the following formulas:

$$\begin{cases} Y_l^m(\theta,\phi) = \sqrt{2} N_l^m P_l^m(\cos\theta)\cos(m\phi), & m > 0 \\ Y_l^{-m}(\theta,\phi) = \sqrt{2} N_l^m P_l^m(\cos\theta)\sin(m\phi), & m > 0 \\ Y_l^0(\theta,\phi) = N_l^0 P_l^0(\cos\theta) \end{cases} \tag{18}$$

We explain all parameters below. l is degree which always greater than 0. m is order, always meet the condition of $-l \le m \le l$.

Figure 4: Qualitative result of GAParts segmentation. We compare our DFGAP with our ablated versions since SOTA methods require 3D inputs that hold different settings from ours. More qualitative results can be found in supplementary materials.

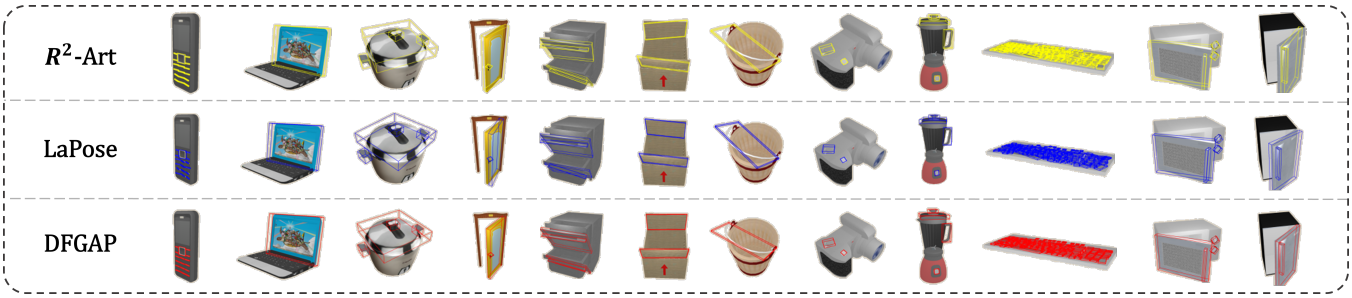

Figure 5: Qualitative results on GAParts pose estimation. More qualitative results can be found in supplementary materials.

$P_l^m(\cos\theta)$ is Associated Legendre polynomials, and N_l^m is an normalized constant. We set a maximum degree $L_{max} = 3$, in that case, there are $(3+1)^2 = 16$ real-valued spherical harmonics for each point p_i^{2D} with angles θ and ϕ. We yield a vector $Y(\theta,\phi) \in \mathbb{R}^{16}$, and, similarly, we apply the vector $Y(\theta,\phi)$ with a learned linear transformation $W_{sh} \in \mathbb{R}^{(d_\theta+d_\rho)\times 16}$ as in:

$$\mathcal{F}_{ang} = W_{sh} \cdot Y(\theta,\phi) \tag{19}$$

to producing a $(d_\theta + d_\phi)$-dimensional angular encoding feature $\mathcal{F}_{ang}$. Finally, the distance encoding $\mathcal{F}_\rho$ and the angular encoding $\mathcal{F}_{ang}$ are concatenated along the feature dimension to yield the overall positional encoding as follow:

$$\mathcal{F}_{kp}^{Enc} = \left[\mathcal{F}_\rho \;||\; \mathcal{F}_{ang}\right] \in \mathbb{R}^{d_\rho+d_\theta+d_\phi} \tag{20}$$

Please note that here d_ρ, d_θ and d_ϕ are hyper-parameters. We set 32 for each parameter and the final dimension is 96. With $\mathcal{F}_{kp}^{Enc}$, we then concatenate it with global feature $\mathcal{F}_k$ as input of pose estimation network.

5 Experiments

5.1 Experiment Setup

Data Preparation We train and evaluate our framework on the GAPartNet dataset [**?**], which contains 9 GAPart classes (such as lids, handles, etc.) across 27 object categories. The dataset provides part-level annotations for 8,489 instances of parts from 1,166 objects, collected from PartNet-Mobility[32] and AKB-48[19].

Evaluation Metrics For part segmentation, we evaluate performance using average precision (AP). Specifically, we use AP50, which is the average precision computed with an Intersection over Union (IoU) threshold of 50%. In addition, we also report the overall average precision (AP) as a supplementary metric. For pose estimation, we use rotation error (in degrees) and translation error (in centimeters). Particularly, for fair comparison, when calculating the translation error for RGB-based methods, we calculate the pixel offset in the 2D RGB image. Then, taking the ground truth depth, we transform it into an error in centimeters.

Evaluation Metrics The two modules, GAParts segmentation and pose estimation are trained separately in our training schedule. For GAParts segmentation, the size of input RGB image is 800×800. For pose estimation, the input image is resized as 224×224 to meet the requirements of DINOv2. For GAParts segmentation, we use RollingUNet as backbone, codes come from the official implement. For pose estimation we use pretrained vits14 model of DINOv2 as backbone. The total training epoch are 600, 300 for the two modules. Training and validation batch size are 16, 32 respectively. All the experiment are implemented on four NVIDIA GeForce RTX 4090 GPUs with 24GB memory.

5.2 Experiment on GAParts Segmentation

We report the results of DFGAP part segmentation evaluated on the GAPartNet dataset that are illustrated in Table.1. DFGAP exceeds other methods by a large margin in multiple metrics. Compared with the GAPartNet baseline [6] in the categories seen, our DFGAP demonstrates an improvement in the segmentation of the vast majority of part categories, with absolute **7.2%** average AP improvement and **6.7%** average AP50 improvement. In unseen categories, our method achieves **36.2%** and **42.8%** for AP and AP50, absolutely **4.2%** and **5.6%** better than the baseline. In particular, in the categories seen, for the GAParts Slider Drawer and Hinge Knob, our

Table 3: Mixture Probability Distribution Modeling Analysis for GAParts segmentation

	Use MPM	Ln.F.HI	Rd.F.HI	Hg.HI	Hg.Ld	Sd.Ld	Sd.Bn	Sd.Dw	Hg.Dr	Hg.Kb	Avg.AP	Avg.AP50
Seen	✘	85.1	61.7	86.7	78.2	91.7	50.8	82.4	89.3	70.5	69.7	77.3
	✔	90.8	**65.7**	92.1	85.9	95.2	68.7	95.5	91.5	70.1	74.8	83.2
Unseen	✘	46.7	45.2	1.2	48.7	6.9	44.5	53.8	69.3	24.4	31.3	37.8
	✔	50.7	47.2	2.3	53.8	7.3	50.9	69.1	72.6	31.8	36.2	42.8

Results of Part Segmentation in terms of Per-GAPart-class AP50 (%), Average AP50 (%), and Average AP (%). Ln.=Line. F.=Fixed. Rd.=Round. Hg.=Hinge. Hl.=Handle. Sd.=Slider. Ld.=Lid. Bn.=Button. Dw.=Drawer. Dr.=Door. Kb.=Knob.

method performs better than the baseline with more than relative **30%** average AP50 improvement (**42.1%** and **32.5%** respectively).

5.3 Experiment on Pose Estimation

The experimental results of pose estimation are presented in Tab.2, our method significantly outperforms the RGB-based baseline methods [13] in the vast majority of GAParts in the GAPartNet dataset. Compared with RGB-D based methods in seen categories, our approach achieves superior performance over the second-best method in most GAPart categories. Specifically, for Round Fixed Handle, Slider Button and Hinge Handle, our approach reduces relative rotation errors by over **40%** (**87.0%**, **67.6%** and **41.1%** respectively). In unseen categories, DFGAP (Ours) outperforms the second-best method in Round Fixed Handle, Slide Lid, and Line Fixed Handle by reducing rotation errors by more than **20%** relative (**83.3%**, **28.9%**, and **21.4%** respectively). For translation, our method also outperforms the GAPartNet baseline [6] on all kinds of GAParts in both seen and unseen categories. The qualitative results are shown in Fig.4 and Fig.5

Table 4: Three Key Components for Pose Estimation Analysis

K.P	D.E	S.H.E	Seen(°)		Unseen(°)	
			Avg.Rot	Avg.Trans	Avg.Rot	Avg.Trans.
✘	✘	✘	14.82	0.214	33.47	0.435
✘	✔	✔	6.77	0.078	18.29	0.177
✔	✔	✘	8.92	0.137	22.17	0.341
✔	✔	✔	**4.74**	**0.021**	**12.59**	**0.076**

The left three columns stand for using self-supervised keypoints prediction, self-supervised depth estimation and spherical harmonic encoding or not. Please note that K.P = keypoints prediction, D.E = depth estimation, S.H.E = spherical harmonic encoding

5.4 Ablation Studies

Mixture Probability Distribution Modeling Analysis

We analyze the effectiveness of our mixture probability modeling (MPM) for all categories GAParts segmentation in Table.3. As can be observed, Remove the module of MPM, for seen categories, the average AP and AP50 absolutely decrease 5.1% and 5.9% respectively. For the unseen category, the average AP and AP50 absolutely decrease 4.9% and 5.0% respectively

Three Key Components for Pose Estimation Analysis

We report the ablation studies for the effect of three key components as shown in Table.4. Compared to the method of removing all three components, our method reduces rotation errors by over 10° in both seen and unseen categories. The experimental result further demonstrate the contribution of each component.

Spherical Harmonic Encoding Analysis

Please refer to Table.5 for quantitative comparison of our designed spherical harmonic encoding for pose estimation. We can have the view that coding in spherical coordinate is better than XYZ coordinate, also, spherical harmonic based coding method is far superior than traditional cosine function based method. In summary, the experimental results demonstrate the well performance of our designed spherical harmonic encoding scheme.

Table 5: Spherical Harmonic Encoding Analysis

Encoding	Coord sys	Seen(°)		Unseen(°)	
		Avg.Rot	Avg.Trans	Avg.Rot	Avg.Trans.
MLP	/	9.71	0.225	23.85	0.385
C.F	XYZ	7.08	0.119	17.75	0.254
C.F	Spherical	6.23	0.064	17.19	0.128
S.H	Spherical	**4.74**	**0.021**	**12.59**	**0.076**

The left two columns stand for different encoding scheme and different coordinate system. Please note that Coord sys is short for coordinate system, and C.F = cosine function encoding, S.H = spherical harmonic encoding.

5.5 Part-based Manipulation Results

We also present the qualitative manipulation results achieved by our framework, along with comprehensive manipulation evaluations conducted in both simulated and real-world environments. We set four different kind of tasks, grasping handle, pushing button, opening drawer and manipulating box. For simulation experiments, we assess the performance using the predicted GAPart segmentation and pose estimation results through SAPIEN [32]. For real-world experiments, we use xArm and its grasp to validate the robustness and precision of our framework in real-world interaction experiments. Due to space limitations, we will give a detailed description of our implement in the supplementary material.

6 Conclusion

In this paper, we propose a novel framework, **DFGAP**, toward precise and robust GAParts segmentation and pose estimation from a single RGB image. From the perspective of reducing the uncertainty in both tasks, we creatively address the depth-free GAPart perception task. Specifically, we independently quantify the uncertainty of segmentation probability for GAParts segmentation and relative depth for pose estimation. Our DFGAP shows superior performance on the GAPartNet dataset compared to various state-of-the-art baseline methods. Multiple sound ablation experiments prove the efficiency and robustness of our proposed methods. Our work has enormous potential in various robot interaction tasks. We firmly believe that our work can be applied in many embodied AI scenarios such as robot manipulation, augmented reality and 3D scene understanding.

7 ACKNOWLEDGEMENTS

This work is supported in part by National Natural Science Foundation of China(NSFC) under Grant 62302143.

References

[1] Yigit Baran Can, Alexander Liniger, Danda Pani Paudel, and Luc Van Gool. 2021. End-to-End Learning of Multi-category 3D Pose and Shape Estimation. *arXiv preprint arXiv:2112.10196* (2021).

[2] Xu Chen, Zijian Dong, Jie Song, Andreas Geiger, and Otmar Hilliges. 2020. Category level object pose estimation via neural analysis-by-synthesis. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16*. Springer, 139–156.

[3] Yamei Chen, Yan Di, Guangyao Zhai, Fabian Manhardt, Chenyangguang Zhang, Ruida Zhang, Federico Tombari, Nassir Navab, and Benjamin Busam. 2024. Secondpose: Se (3)-consistent dual-stream feature fusion for category-level pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9959–9969.

[4] Yan Di, Ruida Zhang, Zhiqiang Lou, Fabian Manhardt, Xiangyang Ji, Nassir Navab, and Federico Tombari. 2022. 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*. 6781–6791.

[5] Zhaoxin Fan, Zhenbo Song, Jian Xu, Zhicheng Wang, Kejian Wu, Hongyan Liu, and Jun He. 2022. Object level depth reconstruction for category level 6d object pose estimation from monocular rgb image. In *European Conference on Computer Vision*. Springer, 220–236.

[6] Haoran Geng, Helin Xu, Chengyang Zhao, Chao Xu, Li Yi, Siyuan Huang, and He Wang. 2023. Gapartnet: Cross-category domain-generalizable object perception and manipulation via generalizable and actionable parts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7081–7091.

[7] Yiran Geng, Boshi An, Haoran Geng, Yuanpei Chen, Yaodong Yang, and Hao Dong. 2023. Rlafford: End-to-end affordance learning for robotic manipulation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 5880–5886.

[8] Dan Guo, Kun Li, Bin Hu, Yan Zhang, and Meng Wang. 2024. Benchmarking micro-action recognition: Dataset, methods, and applications. *IEEE Transactions on Circuits and Systems for Video Technology* 34, 7 (2024), 6238–6252.

[9] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*. 2961–2969.

[10] Junwen Huang, Hao Yu, Kuan-Ting Yu, Nassir Navab, Slobodan Ilic, and Benjamin Busam. 2024. Matchu: Matching unseen objects for 6d pose estimation from rgb-d images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10095–10105.

[11] Li Jiang, Hengshuang Zhao, Shaoshuai Shi, Shu Liu, Chi-Wing Fu, and Jiaya Jia. 2020. Pointgroup: Dual-set point grouping for 3d instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and Pattern recognition*. 4867–4876.

[12] Xiaolong Li, He Wang, Li Yi, Leonidas J Guibas, A Lynn Abbott, and Shuran Song. 2020. Category-level articulated object pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 3706–3715.

[13] Yukang Huo Yan Zhong Jianan Wang Xue Wang Rujing Wang Liu Liu Li Zhang, Haonan Jiang. 2024. R²-Art:Category-level Articulation Pose Estimation from Single RGB Image via Cascade Render Strategy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9959–9969.

[14] Jiehong Lin, Zewei Wei, Changxing Ding, and Kui Jia. 2022. Category-level 6d object pose and size estimation using self-supervised deep prior deformation networks. In *European Conference on Computer Vision*. Springer, 19–34.

[15] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*. 2980–2988.

[16] Haizhuang Liu, Junbao Zhuo, Chen Liang, Jiansheng Chen, and Huimin Ma. 2024. Affinity3D: Propagating Instance-Level Semantic Affinity for Zero-Shot Point Cloud Semantic Segmentation. In *Proceedings of the 32nd ACM International Conference on Multimedia*. 9019–9028.

[17] Liu Liu, Jianming Du, Hao Wu, Xun Yang, Zhenguang Liu, Richang Hong, and Meng Wang. 2023. Category-level articulated object 9d pose estimation via reinforcement learning. In *Proceedings of the 31st ACM International Conference on Multimedia*. 728–736.

[18] Liu Liu, Anran Huang, Qi Wu, Dan Guo, Xun Yang, and Meng Wang. 2024. 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. 3684–3692.

[19] Liu Liu, Wenqiang Xu, Haoyuan Fu, Sucheng Qian, Qiaojun Yu, Yang Han, and Cewu Lu. 2022. Akb-48: A real-world articulated object knowledge base. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 14809–14818.

[20] Liu Liu, Ran Zhang, Wenbo Xu, Li Zhang, Yiming Tang, Qi Wu, and Hao Wu. 2025. GASEM: Boosting Generalized and Actionable Parts Segmentation and Pose Estimation via Object Motion Perception. In *IEEE International Conference on Multimedia Expo*.

[21] Minghua Liu, Yinhao Zhu, Hong Cai, Shizhong Han, Zhan Ling, Fatih Porikli, and Hao Su. 2023. Partslip: Low-shot part segmentation for 3d point clouds via pretrained image-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 21736–21746.

[22] Xueyi Liu, Xiaomeng Xu, Anyi Rao, Chuang Gan, and Li Yi. 2022. Autogpart: Intermediate supervision search for generalizable 3d part segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 11624–11634.

[23] Yutong Liu, Haijiang Zhu, Mengting Liu, Huaiyuan Yu, Zihan Chen, and Jie Gao. 2024. Rolling-unet: Revitalizing mlp's ability to efficiently extract long-distance dependencies for medical image segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 3819–3827.

[24] Kaichun Mo, Leonidas J Guibas, Mustafa Mukadam, Abhinav Gupta, and Shubham Tulsiani. 2021. Where2act: From pixels to actions for articulated 3d objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 6813–6823.

[25] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. 2023. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193* (2023).

[26] Sida Peng, Yuan Liu, Qixing Huang, Xiaowei Zhou, and Hujun Bao. 2019. Pvnet: Pixel-wise voting network for 6dof pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 4561–4570.

[27] Jonas Schult, Francis Engelmann, Alexander Hermans, Or Litany, Siyu Tang, and Bastian Leibe. 2023. Mask3d: Mask transformer for 3d semantic instance segmentation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 8216–8223.

[28] Chen Song, Jiaru Song, and Qixing Huang. 2020. Hybridpose: 6d object pose estimation under hybrid representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 431–440.

[29] Thang Vu, Kookhoi Kim, Tung M Luu, Thanh Nguyen, and Chang D Yoo. 2022. Softgroup for 3d instance segmentation on point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2708–2717.

[30] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J Guibas. 2019. 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*. 2642–2651.

[31] Jiaxin Wei, Xibin Song, Weizhe Liu, Laurent Kneip, Hongdong Li, and Pan Ji. 2024. RGB-based category-level object pose estimation via decoupled metric scale recovery. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2036–2042.

[32] Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao Jiang, Yifu Yuan, He Wang, et al. 2020. Sapien: A simulated part-based interactive environment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 11097–11107.

[33] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. 2017. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199* (2017).

[34] Lumin Xu, Sheng Jin, Wang Zeng, Wentao Liu, Chen Qian, Wanli Ouyang, Ping Luo, and Xiaogang Wang. 2022. Pose for everything: Towards category-agnostic pose estimation. In *European conference on computer vision*. Springer, 398–416.

[35] Qiaojun Yu, Ce Hao, Xibin Yuan, Li Zhang, Liu Liu, Yukang Huo, Rohit Agarwal, and Cewu Lu. 2025. Generalizable Articulated Object Perception with Superpoints. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 1–5.

[36] Xueyu Yuan, Jiabin Fang, Ruoyu Chen, Yukun Zhuang, Li Zhang, Yanyan Wei, and Liu Liu. 2025. Generalizable and Actionable Part Segmentation and Pose Estimation via Cascade Shape-Sensitive Training for Object Manipulation. In *arXiv preprint arXiv*.

[37] Jiyao Zhang, Mingdong Wu, and Hao Dong. 2023. Generative category-level object pose estimation via diffusion models. *Advances in Neural Information Processing Systems* 36 (2023), 54627–54644.

[38] Li Zhang, Zean Han, Yan Zhong, Qiaojun Yu, Xingyu Wu, Xue Wang, and Rujing Wang. 2024. Vocapter: Voting-based pose tracking for category-level articulated object via inter-frame priors. In *Proceedings of the 32nd ACM International Conference on Multimedia*. 8942–8951.

[39] Li Zhang, Zean Han, Yan Zhong, Qiaojun Yu, Xingyu Wu, Xue Wang, and Rujing Wang. 2024. Vocapter: Voting-based pose tracking for category-level articulated object via inter-frame priors. In *Proceedings of the 32nd ACM International Conference on Multimedia*. 8942–8951.

[40] Ruida Zhang, Ziqin Huang, Gu Wang, Chenyangguang Zhang, Yan Di, Xingxing Zuo, Jiwen Tang, and Xiangyang Ji. 2024. LaPose: Laplacian Mixture Shape Modeling for RGB-Based Category-Level Object Pose Estimation. In *European Conference on Computer Vision*. Springer, 467–484.