IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

PartPose: Attentive 6D Pose Estimation by Focusing on Graspable Parts of Multi-Part Deformable Objects

Ryo Okumura[1] and Tadahiro Taniguchi[2,3,4]

***Abstract*—This study tackles robotic picking of multi-part deformable objects—common in warehouses yet underexplored in the literature—such as cable-attached appliances and pouch drinks, which comprise both rigid and deformable components. Their deformability poses a challenge to model-based 6D pose estimators, such as FoundationPose, that assume rigid bodies. To address this, we present PartPose, which estimates the 6D pose of the multi-part deformable objects by focusing on the rigid components. PartPose uses Bayesian optimization to select an appropriate region of interest (ROI) and then estimates its pose with a render-and-compare pipeline. We evaluate pose-estimation and picking success rates on nine multi-part deformable objects, counting a pose estimate as successful if the translational error is ≤30 mm and the rotational error is ≤0.3 radians. PartPose significantly outperforms a FoundationPose baseline, achieving success rates of 98.2% (translational), 96.4% (rotational), and 87.2% (picking), versus 47.9%, 35.9%, and 22.8%, respectively. Moreover, PartPose generalizes category-level semantic knowledge to new instances within the same category without performance degradation when those instances have semantically similar components. This capability is crucial for large logistics centers that handle diverse and novel objects.**

I. Introduction

In 6D pose estimation, objects can be categorized along two axes: single-part versus multi-part, and rigid versus deformable (Table I). Our focus is on multi-part deformable objects, which consist of heterogeneous components, including both rigid and deformable parts. Common examples include items with cables, flexible items with rigid labels, and pouch drinks—all prevalent in logistics warehouses. Despite their widespread presence, these objects have been largely overlooked in robotic picking research.

Conventional 6D pose estimation methods primarily focus on rigid objects and often do not account for the deformability of objects. For example, model-based 6D pose estimation infers the poses of single-part rigid objects by comparing real-world images [1], point clouds [2], or both [3], [4] to 3D models. Model-free 6D pose estimation does not require 3D models; instead, it estimates an object's relative pose in an input image with respect to reference image(s) [5]–[7]. Some studies first generate 3D models from reference images and then estimate pose using model-based approaches [1], [8].

TABLE I
Object classification in the field of 6D pose estimation

	Rigid	Deformable
Single-part	Rigid objects [1]–[8]	Deformable objects [9]–[12]
Multi-part	Articulated objects [13]–[16]	Multi-part deformable objects (Ours)

Multi-part rigid objects, also known as articulated objects, consist of rigid components connected by joints. Existing methods [13]–[16] estimate the pose of each component and infer joint states subject to kinematic constraints. These methods assume that each component is rigid when estimating pose.

All of the above methods assume that objects are rigid and often fail when an object's shape or appearance deviates significantly from the reference 3D models or images. For example, FoundationPose [1], a state-of-the-art model-based 6D pose estimator, can yield significant errors when there is a notable discrepancy between actual objects and their 3D models (Fig. 1(a)).

Zero-shot pose estimation of single-part deformable objects such as cloth and rope remains an open problem. Existing frameworks require object-specific training, such as learning a manipulation policy for each object via reinforcement learning [9], [10], or performing interpolation-based planning in a learned latent space [11], [12]. This poses challenges for manipulating unknown objects in logistics warehouses that handle thousands of products.

To address these challenges, we propose PartPose, a method for estimating 6D poses for unknown multi-part deformable objects by focusing on a specific non-deformable part. Fig. 1(b) highlights the differences between PartPose and FoundationPose. PartPose targets a region of interest (ROI) within the object, rather than the entire object, as FoundationPose does. For instance, PartPose identifies the non-deformable housing of a power strip as the ROI and matches real and rendered images around this area. This allows for accurate 6D pose estimation and grasping while ignoring the cable portion. In contrast, FoundationPose compares the entire object, leading to significant errors due to mistakenly matching the cable. Similarly, for pouch drinks, PartPose accurately estimates the pose of the non-deformable cap by treating it as the ROI, whereas FoundationPose incurs large errors by comparing the entire object.

[1]Ryo Okumura is with Panasonic Connect Co., Ltd., Osaka, Japan. okumura.ryo001@jp.panasonic.com

[2]Tadahiro Taniguchi is with Panasonic Holdings Corp., Osaka, Japan.

[3]Tadahiro Taniguchi is also with Graduate School of Informatics, Kyoto University, Kyoto, Japan. taniguchi@i.kyoto-u.ac.jp

[4]Tadahiro Taniguchi is also with Research Organization of Science and Technology, Ritsumeikan University, Shiga, Japan.

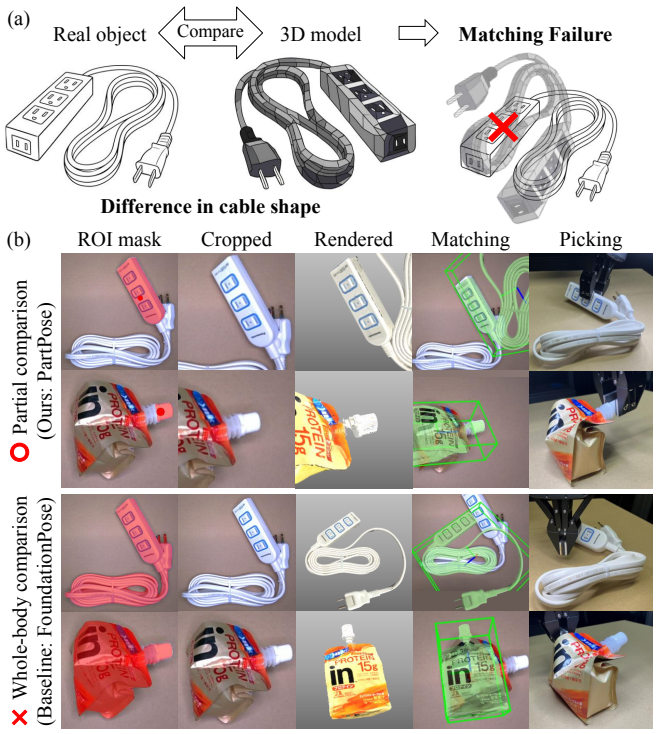


Fig. 1. 6D pose estimation and robotic picking of multi-part deformable objects: (a) Differences in shape between real-world objects and their 3D models can lead to significant errors in pose estimation. (b) Pose estimation through partial comparison (ours) versus whole-body comparison (baseline). In the leftmost column, the ROI mask is represented in red. In the fourth column, rendered images are overlaid on real-world images and displayed in green.

To achieve this, PartPose identifies appropriate ROIs in both real images and 3D models. First, to locate the ROI in the real image, we transfer a predefined keypoint in a reference image to the input image via semantic correspondence. Next, to determine the ROI in the 3D model, we use Bayesian optimization to learn the ROI's center coordinates and spatial extent within the model. Finally, using the obtained ROI information, we crop both the real image and rendered views of the 3D model and estimate the pose using a render-and-compare pipeline. This enables PartPose to leverage existing model-based estimators that assume rigidity while estimating the 6D pose of multi-part deformable objects.

This paper contributes to the field of robotic picking by addressing the challenge of estimating 6D poses for multi-part deformable objects. We propose PartPose, a method that estimates the 6D pose of objects by focusing on specific non-deformable parts suitable for grasping. Furthermore, we demonstrate that PartPose generalizes category-level semantic knowledge to new instances within the same category without performance degradation.

II. FoundationPose

This section provides an overview of FoundationPose, a key technology integral to our approach. FoundationPose is a state-of-the-art model-based 6D pose estimation technique that utilizes a foundation model trained on synthetic data generated by the NVIDIA Isaac Sim robot simulator [17]. In our work, we employ the pre-trained FoundationPose model, which estimates object poses through three primary steps:

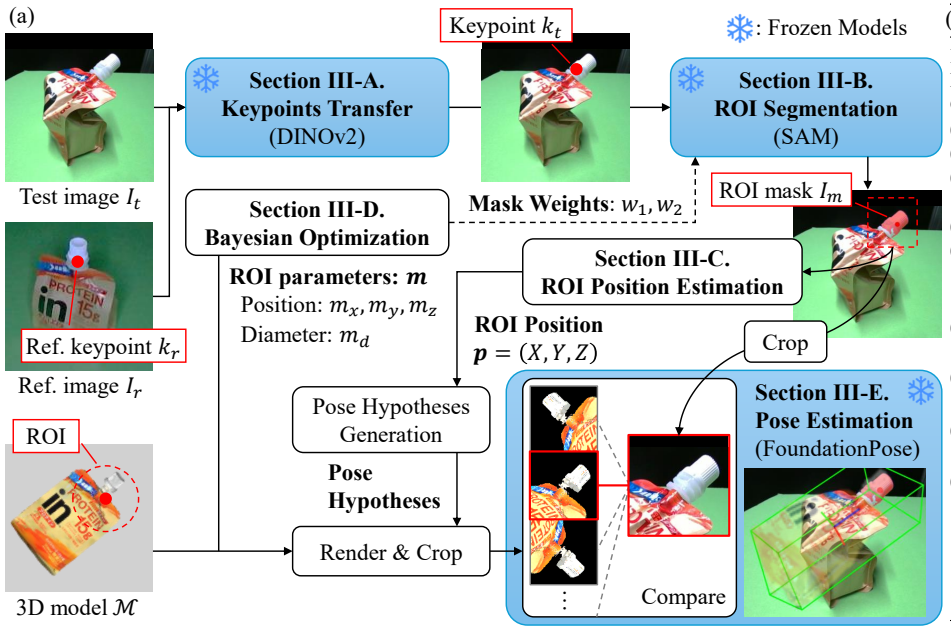
1) Initialization of Pose Hypotheses: FoundationPose begins by detecting and segmenting target objects within input images using either Mask-RNN [18] or CNOS [19], resulting in a 2D bounding box that encloses the segmented area. A preliminary estimate of the object's translational position is derived from RGBD images, using the 3D point corresponding to the median depth within the bounding box. To generate diverse initial camera viewpoints, vertices of an icosphere centered at this 3D point are used, with all viewpoints initially oriented towards the 3D point. These viewpoints are rotated around their optical axes, yielding 252 distinct viewpoints, which serve as initial pose hypotheses.

2) Pose Hypotheses Refinement: Real-world RGB images are cropped using a bounding box centered on the translational position of each pose hypothesis, slightly enlarged to encompass the target object. The bounding box size is based on the 3D model of the target object, enclosing a sphere marginally larger than the diagonal of the 3D bounding box tightly enclosing the model. Pose hypotheses are refined by comparing the rendered 3D model with cropped real-world RGB images. The 3D model is rendered according to each pose hypothesis and cropped similarly to real-world images. Both cropped images are fed into a pose refinement network to predict the difference between the current pose hypothesis and the true pose, adjusting the hypotheses accordingly. This refinement process is iteratively repeated for precise pose estimation.

3) Pose Selection: Finally, refined pose hypotheses are used to render the 3D model, and real-world RGB images are cropped as before. These image pairs are fed into a pose ranking network, which selects the pose hypothesis with the highest score as the final estimated pose of the target object.

III. Method

This section outlines the pose estimation process of PartPose, which employs an iterative render-and-compare strategy similar to FoundationPose but optimizes the ROI to focus on specific object parts. Fig. 2(a) presents an overview of the PartPose process. The process begins with transferring a single reference keypoint from a reference image to a test image using semantic correspondence (Section III-A). This keypoint serves as a prompt for segmenting the focused part using the Segment Anything Model (SAM) [20] (Section III-B). An initial rough estimate of the translational position $\mathbf{p} = (X, Y, Z)$ of this focused part is then derived from the segmented region (Section III-C). The real-world RGB image is cropped around this estimated position. The 3D model $\mathcal{M}$ of the target object is rendered from multiple viewpoints centered on this position, with ROI parameters determined through Bayesian optimization (Section III-D). Finally, both the real-world and rendered images are processed by



(b) **Algorithm 1:** ROI parameter optimization

Input: Test image I_t^i, Test depth image I_d^i, Ref. image I_r, Ref. keypoint k_r, 3D model $\mathcal{M}$, Camera parameters K, Camera pose ${}^{w}T_{ci}$

for $n = 1,2, \ldots$ **do**
  Fitting TPE model with the dataset $\mathcal{D}$
  Select ROI parameters $\boldsymbol{m}$ that maximizes acquisition function.
  for $i = 1,2$ **do**
    Transfer ref. keypoint k_r from ref. image I_r to test image I_t^i and generate new keypoint k_t^i (Sec. III-A.)
    Generate ROI mask I_m^i from test image I_t^i and keypoint k_t^i (Sec. III-B.)
    Estimate ROI position $\boldsymbol{p^i}$ from I_m^i, I_d^i, K (Sec. III-C.)
    Estimate object pose ${}^{ci}T_o$ from I_t^i, $\boldsymbol{p^i}$, $\mathcal{M}$, $\boldsymbol{m}$ (Sec. III-E.)
  end
  Compute objective function (Sec. III-D.) $y = f({}^{c1}T_o, {}^{c2}T_o, {}^{w}T_{c1}, {}^{w}T_{c2})$
  Add the new $(\boldsymbol{m}, y)$ to the dataset $\mathcal{D}$
end
return Optimal ROI parameters $\boldsymbol{m}^*$

Fig. 2. (a) Procedure for 6D pose estimation in PartPose. First, a single keypoint is transferred to a test image from a reference image (Section III-A). Second, the transferred keypoint is utilized to generate a partial mask of ROI for a target object (Section III-B). Mask weights (w_1, w_2) can be used to control the mask areas. Third, an initial translational position $\mathbf{p} = (X, Y, Z)$ is estimated using the partial mask (Section III-C). Pose hypotheses are initialized using the translational position. Fourth, a 3D model $\mathcal{M}$ of the target object is rendered from multiple viewpoints based on the pose hypotheses and ROI parameters $\mathbf{m} = (m_x, m_y, m_z, m_d)$ learned through Bayesian optimization (Section III-D). Finally, an object pose is estimated by comparing the ROI of the test image and the rendered images (Section III-E). (b) Pseudocode for ROI parameter optimization.

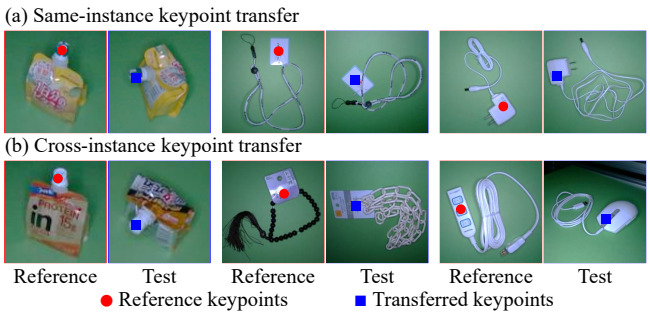


Fig. 3. Keypoints transfer by semantic correspondence. (a) Same-instance keypoints transfer. (b) Cross-instance keypoints transfer.

the pre-trained pose refinement and pose ranking networks of FoundationPose to obtain the final pose estimation values (Section III-E). Each step is detailed below.

A. Keypoints Transfer

First, a single keypoint is annotated within the focused area of the reference image. We calculate the DINOv2 [21] feature map of a test image and transfer the annotated keypoint to the point in the test image where its feature is most similar. Fig. 3 illustrates examples of this keypoint transfer, showing the reference images alongside the target test images. Fig. 3(a) shows same-instance keypoint transfer, where keypoints are transferred from different images of the same object. Fig. 3(b) demonstrates cross-instance keypoint transfer, where keypoints are transferred from a different object within the same category.

B. ROI Segmentation

Using the keypoint obtained in Section III-A as a spatial prompt, we segment the focused area using SAM. We make SAM generate three masks, and adopt the first one. We also explore a method called PartPose-Train-Mask, which combines the three outputs from SAM through weighted averaging, using mask weights $(1-w_1-w_2, w_1, w_2)$, similar to PerSAM [22]. The mask weights (w_1, w_2) are optimized using Bayesian optimization, as detailed in Section III-D.

C. ROI Position Estimation

Based on the partial mask obtained in Section III-B, we estimate the initial translational position $\mathbf{p} = (X, Y, Z)$ of the focused area. To mitigate the impact of small, disconnected fragments in the SAM-generated masks, we extract contours from the partial mask and select the largest connected region. The centroid of this region is used as the ROI center (u, v) in the image coordinate system. We estimate the distance Z from the camera to the object by averaging the depth values within this region. Using the ROI center (u, v), the estimated distance Z, and the camera's intrinsic parameters K, we compute the 3D translational position $\mathbf{p}$ in the camera coordinate system.

D. Bayesian Optimization

We optimize the ROI parameters $\mathbf{m}$ which consist of the ROI origin offset (m_x, m_y, m_z) and the ROI diameter m_d using Bayesian optimization. The ROI parameters $\mathbf{m}$ are normalized based on the dimensions of each side and the diagonal length of the 3D bounding box enclosing the 3D model $\mathcal{M}$, resulting in normalized values within $[0, 1]$. This normalization scales the ROI parameters according to the object's size. In PartPose-Train-Mask, we also optimize the SAM mask weights (w_1, w_2). To ensure the combined

weight does not exceed 1, we normalize the weights such that $w_1 + w_2 = 1$ if their initial sum is greater than 1.

To calculate the objective function, we assume that the object is static and take two images from known camera poses, ${}^{w}T_{c_1}$ and ${}^{w}T_{c_2}$, defined in the world coordinate system. These poses can be determined if the camera is mounted on a robotic arm with known kinematics or if the camera setup is pre-calibrated and fixed. Next, the object poses in the camera coordinate system, ${}^{c_1}T_o$ and ${}^{c_2}T_o$, are estimated as explained in Section III-E. We calculate the object poses in the world coordinate system as ${}^{w}T_{c_1}\,{}^{c_1}T_o$ and ${}^{w}T_{c_2}\,{}^{c_2}T_o$, and compare them. The translational distance is measured by the Euclidean distance between the translation vectors, and the 3D rotational distance is calculated as $2 \times \arccos(|q_1 \cdot q_2|)$ [23], [24], where q_1 and q_2 are the quaternions representing the 3D rotations. We compute a weighted sum of these distances, with weights of w_t=1.0 for translation (in meters) and w_r=0.1 for rotation (in radians). Robustness to these parameters is evaluated in Section IV-G. We minimize this objective function using Bayesian optimization to find the optimal values for the trainable parameters.

E. Pose Estimation

We use the translational position $\mathbf{p} = (X, Y, Z)$ obtained in Section III-C to initialize pose hypotheses, following the procedure of FoundationPose. We render the 3D model $\mathcal{M}$ based on these initial hypotheses. In the 3D model's coordinate system, we define the ROI origin as a point offset by (m_x, m_y, m_z), and the ROI origin is set to the translational position $\mathbf{p}$ during rendering. We crop both the real-world and rendered images using a bounding box centered around $\mathbf{p}$, ensuring both images focus on the area of interest. This bounding box encloses a sphere with a diameter 1.2 times the ROI diameter m_d. By optimizing the ROI parameters $\mathbf{m} = (m_x, m_y, m_z, m_d)$ through Bayesian optimization, as detailed in Section III-D, we focus on the most informative region to improve pose estimation accuracy. The final pose estimation is achieved by refining the pose hypotheses over five iterations using the pre-trained pose refinement and pose ranking networks from FoundationPose.

IV. Experiments

In this section, we describe experiments designed to demonstrate the effectiveness of PartPose for 6D pose estimation of multi-part deformable objects. Furthermore, to evaluate PartPose's adaptability to unseen objects, we investigate the feasibility of transferring category-level knowledge for reference keypoints and trained parameters. Standard datasets [25]–[30] and metrics [28], [31] used in Benchmark for 6D Object Pose Estimation (BOP) [32], [33] assume that objects are rigid and do not deform. Therefore, we created our own dataset and defined appropriate metrics specifically for multi-part deformable objects.

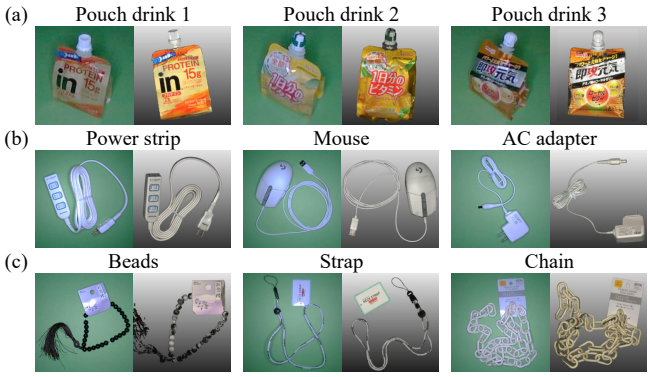


Fig. 4. Target objects (green background) and 3D models (gray background). The objects are categorized into three groups: (a) Pouch drinks, (b) Cable-attached objects, (c) String-like objects.

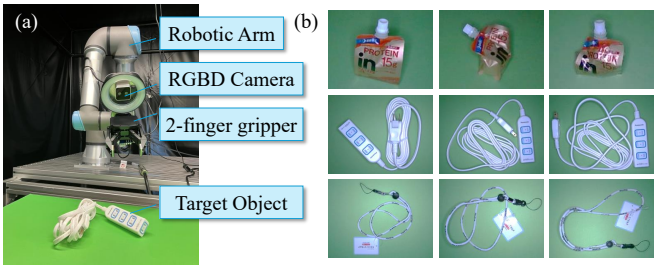


Fig. 5. (a) A robotic arm equipped with an RGBD camera to capture images from various viewpoints. A two-finger gripper for object picking is mounted on the robotic arm. (b) Sample images illustrating the shape variation across different pattern subsets.

A. Data Collection

Fig. 4 illustrates the target objects and the 3D models we created for our dataset. We collected data for three distinct instances within each of the following three categories: (a) pouch drinks, (b) cable-attached objects, and (c) string-like objects. For each individual object, we prepared a single reference image and manually marked a keypoint near the center of its ROI. These designated keypoints were subsequently transferred to each test image. In the experiment evaluating category-level semantic knowledge transfer, the reference keypoints were transferred from a different instance belonging to the same object category.

Fig. 5 (a) depicts the system employed for data collection and object picking. An RGBD camera (RealSense D405) and ring lighting were mounted on a robotic arm (UR5e). A two-finger gripper (ROBOTIS HAND RH-P12-RN), used for picking, is also mounted on the robotic arm. During data collection, the target object was positioned beneath the camera, and RGBD images were captured from various viewpoints at an approximate distance of 30 cm. We deformed the shape of the objects' flexible parts to generate multiple distinct dataset patterns. We refer to each partial dataset generated from these variations as a pattern subset. Fig. 5 (b) shows representative sample images. In this example, the shape of the pouch drink, as well as the configuration of the cable of the power strip and the string of the strap, varies across the different pattern subsets. We generated five training pattern subsets, each containing images from eight viewpoints, and five test

pattern subsets, each with images from 27 viewpoints. The camera poses during image acquisition were recorded using the robot's self-localization system and are included in the dataset. These recorded camera poses are utilized to compute the objective function in the Bayesian optimization process. The 3D models were created using a 3D scanner (KEYENCE VL-700).

B. Optimization

We applied Bayesian optimization as described in Section III-D to determine the optimal values for the ROI parameters $\mathbf{m} = (m_x, m_y, m_z, m_d)$. Fig. 2(b) shows the pseudocode for the ROI parameter optimization process. The training procedure utilized images captured from eight different viewpoints across the five training pattern subsets. These eight images were randomly grouped into four pairs, resulting in a total of 20 pairs across all five training pattern subsets. For each of these pairs, we calculated the objective function. We employed the Tree-structured Parzen Estimator (TPE) algorithm [34] using the Optuna optimization framework [35]. We adopted TPE with a view toward future extensions of our proposed method for optimizing discrete parameters, such as labels to select the optimal ROI from multiple candidates, as will be discussed in Section V. We empirically found that 200 optimization trials were sufficient to adequately optimize the parameters. In most cases, the optimization converged within 50 trials. The training time was approximately 3.3 hours using a single NVIDIA RTX 3090 Ti GPU.

C. Evaluation

We conducted an evaluation of PartPose and compared to baselines and an ablation below.

PartPose (Ours): The proposed method that optimizes the ROI parameters $\mathbf{m}$.

PartPose-Train-Mask (Ours): A variant of PartPose that additionally optimizes the mask-weight parameters w_1 and w_2, following PerSAM.

PerSAM-FoundationPose: A baseline that generates target masks using PerSAM and estimates 6D poses with FoundationPose. First, PerSAM derives a feature vector from a reference image and its corresponding mask. This feature vector is then used to locate the point in the test image that most closely resembles the target object. This identified point serves as a spatial prompt for SAM to generate the mask in the test image.

SAM-6D [3]: The state-of-the-art model-based pose estimation method equipped with a partial-to-partial correspondence mechanism that ignores parts of the scene unrelated to the target object. We evaluate whether this mechanism is effective for pose estimation of multi-part deformable objects.

Any6D [8]: The state-of-the-art model-free pose estimation method that utilizes 3D models generated from a single reference image and estimates 6D poses with FoundationPose.

Training-Free-PartPose: An ablation variant of PartPose without optimization of ROI parameters.

Due to significant shape variations between the 3D models and real-world objects, standard 6D pose estimation metrics like average distance [28] and visual surface discrepancy [31] are not applicable. Instead, we compared the estimated and ground-truth poses and computed translational and rotational errors.

Our primary evaluation metric was the success rate, defined as the percentage of estimations with a translation error within 30 mm and a rotation error within 0.3 radians. We estimated the errors for 135 images per object, covering 27 different viewpoints across five test pattern subsets, and calculated the average success rate for each object category.

Ground truth poses were determined by averaging five pose estimation results that were visually verified as correct by human annotators. Correct estimations were identified by overlaying the rendered 3D model onto the real-world RGB image, allowing a human to visually assess and judge the accuracy of the alignment. To validate the annotation method, five participants annotated the ground truth poses for three objects: a pouch drink 2, a power strip, and beads. The translational and rotational variations of the annotated poses were, on average, 2 mm and 0.03 radians, respectively, with maximums of 6 mm and 0.07 radians. Compared to the threshold for successful pose estimation, the average variation was an order of magnitude smaller. This result validates the soundness of our ground-truth pose annotation method.

Additionally, we evaluated the success rate of an object picking task using PartPose and PerSAM-FoundationPose. A two-finger gripper mounted on the robotic arm (Fig. 5(a)) was used to grasp each object based on its estimated pose. We predefined desired grasp poses in the 3D model's coordinate system (${}^{o}T_{g}$), then estimated the object poses in the camera coordinate system (${}^{c}T_{o}$), and finally computed the gripper pose in the world coordinate system as: ${}^{w}T_{g}={}^{w}T_{c}\,{}^{c}T_{o}\,{}^{o}T_{g}$. The grasp pose ${}^{o}T_{g}$ was defined by taking the center of the object's rigid part as the grasp point and orienting the gripper so that its two fingers were parallel to either the shortest or the second-shortest axis of the bounding box enclosing that rigid part. For certain objects, such as the strap, the rigid part was too thin, so we employed grasp poses oriented only parallel to the shortest axis. For each object, we performed 20 grasp attempts while varying the object's pose and the configuration of its flexible part.

D. Success Rate of 6D Pose Estimation and Picking

Table II presents the experimental results. In the average success rates across all objects (the rightmost column of the table), PartPose achieved the highest translational (98.2%) and rotational (96.4%) success rates, significantly outperforming the baselines. The baselines struggled, especially with string-like objects due to their large deformation, with PerSAM-FoundationPose achieving much lower success rates of translation (29.9%) and rotation (25.2%).

TABLE II
SUCCESS RATE OF POSE ESTIMATION AND PICKING. THE HIGHEST PERFORMANCES ARE INDICATED WITH UNDERLINES.

Translation	Pouch	Cable	String	**All**
PartPose(Ours)	96.3%	99.3%	99.0%	98.2%
PartPose-Train-Mask(Ours)	95.8%	97.5%	99.0%	97.4%
PerSAM-FoundationPose	44.7%	69.1%	29.9%	47.9%
SAM-6D	68.4%	62.5%	26.7%	52.5%
Any6D	40.0%	4.0%	20.7%	21.6%
Training-Free-PartPose	36.5%	50.4%	35.8%	40.9%
Rotation	Pouch	Cable	String	**All**
PartPose(Ours)	91.1%	99.0%	99.0%	96.4%
PartPose-Train-Mask(Ours)	90.4%	97.3%	99.0%	95.6%
PerSAM-FoundationPose	24.7%	57.8%	25.2%	35.9%
SAM-6D	42.7%	54.8%	20.3%	39.3%
Any6D	7.2%	1.2%	22.0%	10.1%
Training-Free-PartPose	21.0%	43.2%	30.4%	31.5%
Picking	Pouch	Cable	String	**All**
PartPose(Ours)	83.3%	95.0%	83.3%	87.2%
PerSAM-FoundationPose	26.7%	38.3%	3.3%	22.8%

Figure 1(b) provides examples of the ROI mask, cropped real-world and rendered images, and their overlay along the estimated poses using PartPose and the baseline method, PerSAM-FoundationPose. With PartPose, the ROI masks are focused on non-deformable regions, which allows for accurate pose estimation. In contrast, PerSAM-FoundationPose applies masks to the entire object, leading to significant errors. For the power strip, PerSAM-FoundationPose attempts to match the shape of the cable, which is flexible and prone to deformation. For the pouch drink, PerSAM-FoundationPose struggles to match the entire body, which has a variable shape in the real-world and rendered images. These mismatches lead to significant errors in pose estimation, as PerSAM-FoundationPose fails to ignore shape differences. By focusing on stable, non-deformable parts, PartPose achieves more reliable pose estimations than the baseline approach.

PartPose-Train-Mask exhibited slightly lower performance than PartPose, but the difference was marginal. Across all objects, PartPose and PartPose-Train-Mask achieved translational errors of 7.4 ± 12.2 and 8.4 ± 17.8 mm, and rotational errors of 0.2 ± 0.4 and 0.2 ± 0.5 radians, respectively. A t-test at the 5% significance level showed no significant difference. These results suggest that PartPose's strategy of using the first mask output of SAM is optimal, and that there is no need to adjust mask weights as in PartPose-Train-Mask.

SAM-6D achieved a slightly higher performance than the other baselines, but its pose estimation success rates were still low (52.5% for translation and 39.3% for rotation). This suggests that the partial-to-partial correspondence mechanism of SAM-6D is not effective in scenarios involving the deformation of flexible parts. Because the partial-to-partial correspondence relies on image feature similarity, it can ignore regions with different appearances (e.g., background or other objects), but it cannot accommodate deformations of the target object itself.

Any6D achieved the lowest performance, with translational (21.6%) and rotational (10.1%) success rates, due to a 3D modeling error. Since the single reference image does not contain information about the object's back side, the rotational estimation accuracy was extremely low.

Training-Free-PartPose showed a significantly lower performance than the proposed method, with translational and rotational success rates of 40.9% and 31.5%, respectively. This suggests that ROI parameter optimization plays a crucial role in PartPose.

The picking success rates followed the same trend as the pose estimation. PartPose achieved an 87.2% success rate, significantly outperforming PerSAM-FoundationPose (22.8%), which struggled particularly with string-like objects, achieving only a 3.3% success rate. These results suggest that models with a higher pose estimation success rate also have a higher picking success rate.

TABLE III
SUCCESS RATE OF POSE ESTIMATION USING CROSS-INSTANCE KNOWLEDGE TRANSFER IN PARTPOSE.

Translation	Pouch	Cable	String
w/o knowledge transfer	96.3%	99.3%	99.0%
Semantic knowledge transfer	94.1%	51.1%	93.0%
Geometric knowledge transfer	86.7%	-	-
Combined knowledge transfer	87.8%	-	-
Rotation	Pouch	Cable	String
w/o knowledge transfer	91.1%	99.0%	99.0%
Semantic knowledge transfer	87.4%	48.9%	92.6%
Geometric knowledge transfer	60.4%	-	-
Combined knowledge transfer	62.6%	-	-

E. Results for Category-level Knowledge Transfer

We explored how PartPose facilitates category-level knowledge transfer by transferring information from a source object to a target object within the same category. We selected pouch drink 1, power strip, and beads as source objects and transferred knowledge to two other target instances in each category. We evaluated pose estimation performance through three strategies of knowledge transfer: *semantic knowledge transfer*, *geometric knowledge transfer*, and *combined knowledge transfer*.

In *semantic knowledge transfer*, reference keypoints from the source object were transferred to corresponding location in target object images, as shown in Fig. 3 (b), reducing the need for keypoint annotation on reference images. For *geometric knowledge transfer*, we pre-trained the ROI parameters of the source object and applied them to the target object. In *combined knowledge transfer*, both the reference keypoints and the ROI parameters were transferred from the source to the target object. Note that the ROI parameters cannot be transferred in non-standardized products like cable-attached and string-like objects, where ROI positions vary among instances.

Table III shows average success rates for translation and rotation across the three strategies. For comparison, we included results for *w/o knowledge transfer*, representing standard PartPose's performance without transferring knowledge from the source objects.

For the pouch drinks and string-like objects, *semantic knowledge transfer* achieved competitive performance to

the *w/o knowledge transfer* method, whereas performance for the cable-attached objects degraded substantially. We attribute this to the fact that the pouch drinks and string-like objects contain semantically similar parts, such as plastic caps or paper labels, which meant keypoint transfer hardly failed. In contrast, when transferring keypoints from the power strip to the mouse, we found that 86% of the keypoints were mapped to the mouse's USB plug. This factor is responsible for almost all observed failures in *semantic knowledge transfer*.

On the other hand, with *geometric knowledge transfer*, the translational and rotational success rates fell markedly to 86.7% and 60.4%, respectively. In theory, ROI parameters can be transferred as long as the geometrical shapes are the same; however, in practice, we found that object deformations in the 3D models caused the pose of ROI parts to vary from one instance to another. Successful ROI parameter transfer depends on the geometric consistency of 3D models across instances. For deformable objects, maintaining shape consistency in 3D-scanned models is difficult, and highly accurate models, such as those produced by CAD, may be required.

F. Inference Speed

We measured the time required by PartPose for pose estimation and tracking. Since the render-and-compare process is influenced by the rendering speed of 3D models, we report the average and standard deviation for the pose hypothesis refinement and ranking across nine objects. All experiments were conducted on an Intel i9-14900K CPU and an NVIDIA RTX 3090 Ti GPU. PartPose took 0.2 seconds for keypoint transfer, 0.02 seconds for ROI segmentation, and 1.7±0.3 seconds for pose hypothesis refinement and ranking, for a total of around 1.9 seconds. Compared to FoundationPose, there is no additional computational cost apart from the keypoint transfer. For pose tracking, we followed the same approach as FoundationPose by using the pose estimated in the previous step as the pose hypothesis. This eliminates the need for keypoint transfer and ROI segmentation, and it reduces the number of pose hypothesis candidates to a single one. We refined the pose hypothesis twice. Under these conditions, the pose tracking time was 0.014 seconds, making real-time pose tracking feasible.

G. Robustness to Weights in Objective Function

As described in Section III-D, PartPose uses the weighted sum of translational and rotational errors as its objective function, with weights w_t and w_r. To evaluate the training robustness to these weights, we fixed w_t=1.0 and varied w_r. We chose one object from each category (pouch drink 1, AC adapter, and strap) and evaluated the pose estimation accuracy (Table IV). A t-test at the 5% significance level showed no significant difference between the case of w_r=0.1 and the other settings. These results suggest that PartPose's training is robust to the choice of objective function weights.

TABLE IV
ROBUSTNESS TO ROTATIONAL ERROR WEIGHT w_r. THE TRANSLATIONAL ERRORS [MM] AND ROTATIONAL ERRORS [RADIANS] ARE SHOWN AS MEAN ± STANDARD DEVIATION.

w_r	0.01	0.03	0.1	0.3	1.0
Trans.	6±8	6±4	6±8	7±12	5±4
Rot.	0.2±0.5	0.2±0.5	0.2±0.5	0.2±0.6	0.2±0.5

V. Conclusion

This study has addressed the overlooked challenge of 6D pose estimation for multi-part deformable objects in robotic picking scenarios. By introducing PartPose, we have provided a robust solution that focuses on the non-deformable components of the multi-part deformable objects, enabling accurate pose estimation of the graspable parts. Our research highlights the limitations of conventional model-based 6D pose estimation methods, which predominantly target rigid objects and often result in significant errors when applied to deformable items. PartPose overcomes these limitations by identifying and concentrating on specific ROIs within the objects, particularly the graspable, non-deformable parts, significantly reducing errors associated with the deformable components. Moreover, we have demonstrated that category-level knowledge transfer can facilitate 6D pose estimation for novel objects. In particular, semantic knowledge transfer via cross-instance keypoint transfer reduces the effort required to annotate keypoints on reference images and enhances the adaptability and efficiency of robotic systems in large-scale logistics warehouses.

PartPose has a few limitations. The method isn't suitable for objects that are entirely deformable since it relies on identifying non-deformable parts to accurately estimate the pose. Also, similar to other 6D pose estimation techniques, PartPose's accuracy is reduced when working with objects that have minimal texture or surface features, which are vital for reliable pose estimation.

Several promising directions exist for future work. First, we could improve robustness against various disturbances in real-world warehouse environments. For example, if an ROI region is occluded in cluttered settings or a rigid part like a paper label is deformed, accurate pose estimation becomes difficult. To address this, we can assign multiple ROIs to a single object and select an adequate one that is neither occluded nor deformed.

Furthermore, to enhance resilience to changes in illumination, it would be effective to tune the lighting parameters used during rendering to match the lighting conditions of the 3D model with those of the real environment. For instance, one could learn rendering parameters along with the ROI parameters to acquire lighting settings tailored to each deployment site. Alternatively, one could dynamically adjust lighting parameters during the pose hypothesis refinement process to achieve more precise pose estimates.

Second, improving scalability for deployment in large-scale warehouses is also a critical challenge. For example, PartPose currently estimates ROIs using keypoints annotated on reference images, but it would be desirable to

extend the system to operate without any annotations. One possible direction is to leverage Vision-Language Models [36]–[38] to automatically identify rigid and easily graspable regions in reference images. Alternatively, it is conceivable to automatically detect salient regions and, for each region, learn a label—whether to adopt it as an ROI or not—using Bayesian optimization. Additionally, the requirement of 3D models for model-based 6D pose estimation can become a bottleneck for scalability. One potential extension is toward model-free 6D pose estimation techniques [5]–[8], which could enhance applicability to cases where the object's 3D model is unavailable.

These future directions hold the potential to further enhance the robustness and applicability of PartPose, paving the way for more sophisticated and versatile robotic manipulation systems in large-scale warehouses.

References

[1] B. Wen, W. Yang, J. Kautz, and S. Birchfield, "FoundationPose: Unified 6D pose estimation and tracking of novel objects," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024.

[2] P. J. Besl and N. D. McKay, "Method for registration of 3-D shapes," in *Sensor fusion IV: control paradigms and data structures*, vol. 1611. SPIE, 1992, pp. 586–606.

[3] J. Lin, L. Liu, D. Lu, and K. Jia, "SAM-6D: Segment anything model meets zero-shot 6d object pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024.

[4] A. Caraffa, D. Boscaini, A. Hamza, and F. Poiesi, "Freeze: Training-free zero-shot 6D pose estimation with geometric and vision foundation models," in *Proc. Eur. Conf. Comput. Vis.*, 2024.

[5] J. Liu, W. Sun, K. Zeng, J. Zheng, H. Yang, L. Wang, H. Rahmani, and A. Mian, "Novel object 6D pose estimation with a single reference view," *arXiv preprint arXiv:2503.05578*, 2025.

[6] W. Shi, S. Gai, F. Da, and Z. Cai, "SamPose: Generalizable model-free 6D object pose estimation via single-view prompt," *RA-L*, 2025.

[7] Y. Liu, Y. Wen, S. Peng, C. Lin, X. Long, T. Komura, and W. Wang, "Gen6D: Generalizable model-free 6-DOF object pose estimation from RGB images," in *Proc. Eur. Conf. Comput. Vis.*, 2022.

[8] T. Lee, B. Wen, M. Kang, G. Kang, I. S. Kweon, and K.-J. Yoon, "Any6D: Model-free 6D pose estimation of novel objects," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2025.

[9] T. Hoang, H. Le, P. Becker, V. A. Ngo, and G. Neumann, "Geometry-aware RL for manipulation of varying shapes and deformable objects," in *Proc. Int. Conf. Learn. Represent.*, 2025.

[10] B. Chen, P. Abbeel, and D. Pathak, "Unsupervised learning of visual 3D keypoints for control," in *Proc. Int. Conf. Mach. Learn.*, 2021.

[11] T. Kurutach, A. Tamar, G. Yang, S. J. Russell, and P. Abbeel, "Learning plannable representations with causal infoGAN," in *Adv. Neural Inf. Process. Syst.*, 2018.

[12] P. Zhou, J. Zhu, S. Huo, and D. Navarro-Alarcon, "LaSeSOM: A latent and semantic representation framework for soft object manipulation," *RA-L*, vol. 6, no. 3, pp. 5381–5388, 2021.

[13] X. Li, H. Wang, L. Yi, L. J. Guibas, A. L. Abbott, and S. Song, "Category-level articulated object pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020.

[14] X. Liu, J. Zhang, R. Hu, H. Huang, H. Wang, and L. Yi, "Self-supervised category-level articulated object pose estimation with part-level SE (3) equivariance," in *Proc. Int. Conf. Learn. Represent.*, 2023.

[15] X. Yu, H. Jiang, L. Zhang, L. Y. Wu, L. Ou, and L. Liu, "EfficientCAPER: An end-to-end framework for fast and robust category-level articulated object pose estimation," in *Adv. Neural Inf. Process. Syst.*, 2024.

[16] J. Huang, H. Lin, T. Wang, Y. Fu, Y.-G. Jiang, and X. Xue, "You only estimate once: Unified, one-stage, real-time category-level articulated object 6D pose estimation for robotic grasping," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2025.

[17] Z. Zhou, J. Song, X. Xie, Z. Shu, L. Ma, D. Liu, J. Yin, and S. See, "Towards building AI-CPS with NVIDIA Isaac Sim: An industrial benchmark and case study for robotics manipulation," in *Proc. Int. Conf. Softw. Eng.: Softw. Eng. Pract.*, 2024.

[18] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2017.

[19] V. N. Nguyen, T. Groueix, G. Ponimatkin, V. Lepetit, and T. Hodan, "CNOS: A strong baseline for CAD-based novel object segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023.

[20] A. Kirillov, *et al.*, "Segment anything," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023.

[21] M. Oquab, *et al.*, "DINOv2: Learning robust visual features without supervision," *Trans. Mach. Learn. Res.*, 2024.

[22] R. Zhang, Z. Jiang, Z. Guo, S. Yan, J. Pan, X. Ma, H. Dong, P. Gao, and H. Li, "Personalize segment anything model with one shot," in *Proc. Int. Conf. Learn. Represent.*, 2024.

[23] P. Wunsch, S. Winkler, and G. Hirzinger, "Real-Time pose estimation of 3D objects from camera images using neural networks," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1997.

[24] D. Q. Huynh, "Metrics for 3D rotations: Comparison and analysis," *Journal of Mathematical Imaging and Vision*, vol. 35, pp. 155–164, 2009.

[25] T. Hodan, P. Haluza, Š. Obdržálek, J. Matas, M. Lourakis, and X. Zabulis, "T-LESS: An RGB-D dataset for 6D pose estimation of texture-less objects," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2017.

[26] B. Drost, M. Ulrich, P. Bergmann, P. Hartinger, and C. Steger, "Introducing MVTec ITODD - a dataset for 3D object recognition in industry," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops*, 2017.

[27] R. Kaskman, S. Zakharov, I. Shugurov, and S. Ilic, "HomebrewedDB: RGB-D dataset for 6D pose estimation of 3D objects," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops*, 2019.

[28] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab, "Model based training, detection and pose estimation of texture-less 3D objects in heavily cluttered scenes," in *Proc. Asian Conf. Comput. Vis.*, 2012.

[29] E. Brachmann, A. Krull, F. Michel, S. Gumhold, J. Shotton, and C. Rother, "Learning 6D object pose estimation using 3D object coordinates," in *Proc. Eur. Conf. Comput. Vis.*, 2014.

[30] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, "PoseCNN: A convolutional neural network for 6D object pose estimation in cluttered scenes," in *Proc. Robot.: Sci. Syst.*, 2018.

[31] T. Hodaň, J. Matas, and Š. Obdržálek, "On evaluation of 6D object pose estimation," in *Proc. Eur. Conf. Comput. Vis.*, 2016.

[32] T. Hodan, *et al.*, "BOP: Benchmark for 6D object pose estimation," in *Proc. Eur. Conf. Comput. Vis.*, 2018.

[33] V. N. Nguyen, *et al.*, "BOP challenge 2024 on model-based and model-free 6D object pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2025.

[34] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," in *Adv. Neural Inf. Process. Syst.*, vol. 24, 2011.

[35] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," in *Proc. ACM SIGKDD Conf. Knowl. Discov. Data Min.*, 2019.

[36] W. Yuan, J. Duan, V. Blukis, W. Pumacay, R. Krishna, A. Murali, A. Mousavian, and D. Fox, "RoboPoint: A vision-language model for spatial affordance prediction for robotics," in *Proc. Conf. Robot Learn.*, 2024.

[37] S. Liu, *et al.*, "Grounding DINO: Marrying DINO with grounded pre-training for open-set object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2024.

[38] L. H. Li, *et al.*, "Grounded language-image pre-training," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022.