This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Adaptive Articulated Object Manipulation On The Fly with Foundation Model Reasoning and Part Grounding

Xiaojie Zhang*[1], Yuanfei Wang*[2], Ruihai Wu*[2],
Kunqi Xu[3], Yu Li[1], Liuyu Xiang[1], Hao Dong†[2], Zhaofeng He†[1]
[1] Beijing University of Posts and Telecommunications
[2] School of Computer Science, Peking University [3] School of EECS, Peking University

Abstract

Articulated objects pose diverse manipulation challenges for robots. Since their internal structures are not directly observable, robots must adaptively explore and refine actions to generate successful manipulation trajectories. While existing works have attempted cross-category generalization in adaptive articulated object manipulation, two major challenges persist: (1) the geometric diversity of real-world articulated objects complicates visual perception and understanding, and (2) variations in object functions and mechanisms hinder the development of a unified adaptive manipulation strategy. To address these challenges, we propose ***AdaRPG****, a novel framework that leverages foundation models to extract object parts, which exhibit greater local geometric similarity than entire objects, thereby enhancing visual affordance generalization for functional primitive skills. To support this, we construct a part-level affordance annotation dataset to train the affordance model. Additionally, AdaRPG utilizes the common knowledge embedded in foundation models to reason about complex mechanisms and generate high-level control codes that invoke primitive skill functions based on part affordance inference. Simulation and real-world experiments demonstrate AdaRPG's strong generalization ability across novel articulated object categories.*

1. Introduction

Articulated object manipulation is a fundamental challenge in robotics, requiring robots to interact with objects composed of multiple interconnected parts and joints. Real-world tasks often demand adaptive manipulation, where robots must handle realistic, long-horizon interactions by reasoning about an object's functional mechanisms. Adaptive manipulation requires robots to infer hidden states and adjust their actions accordingly. For example, a safe can only be opened if its latch is unlocked—yet the lock state is invisible to the robot, forcing it to attempt pulling the handle, observe feedback, and adapt its strategy. This adaptability is essential for autonomous robots operating in dynamic, unstructured environments, where object behaviors are not explicitly predefined.

Despite extensive research, cross-category generalization in adaptive articulated object manipulation remains a major challenge due to two key factors. First, geometric diversity among real-world articulated objects complicates visual perception and affordance learning. Traditional visual affordance models struggle to generalize across novel object categories, limiting their ability to transfer functional skills. Second, variation in manipulation mechanisms—including joint constraints, locking mechanisms, and randomized rotations—prevents direct policy transfer between object categories. These factors create a significant barrier to developing a unified adaptive manipulation policy that works across diverse articulated objects.

Fortunately, recent advances in foundation vision and language models have demonstrated strong generalization capabilities in grounding and reasoning. Trained on vast datasets, these models present a promising opportunity to enhance robotic perception and mechanism reasoning. At the same time, articulated objects consist of multiple functional parts, with manipulation trajectories inherently determined by these parts. Moreover, different object categories often share similar parts, making part-based representations a natural and effective choice for articulated object manipulation. Building on these insights, we introduce **Ada**ptive Manipulation with Foundation Model **R**easoning and **P**art **G**rounding (**AdaRPG**), a novel framework that utilizes foundation models to achieve generalizable adaptive articulated object manipulation.

As shown in Fig. 1, we first construct a part affordance dataset, consisting of part-level point clouds and corresponding affordance annotations. During inference, AdaRPG grounds and segments object parts to extract the relevant part point cloud, which is then used to infer part affordance for guiding primitive skill functions. Since object

*Equal contribution. †Corresponding authors.

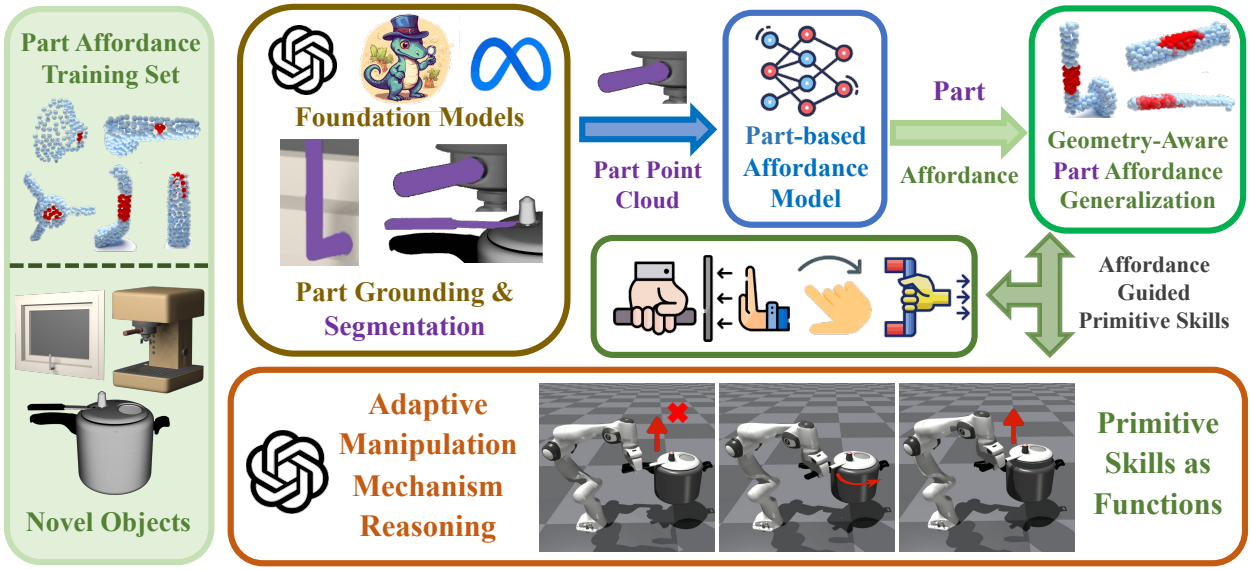

Figure 1. Overview of **AdaRPG**. We construct a part-level affordance dataset to train the affordance model, which takes a part point cloud as input. During inference, AdaRPG first leverages foundation vision and language models to segment the part point cloud and predict part affordance, guiding primitive skills such as grasping and rotating. AdaRPG then utilizes GPT-4o to reason about manipulation mechanisms and generate Python code that invokes primitive skill functions for execution.

parts share greater local geometric similarity than whole objects, this approach significantly improves affordance generalization to novel objects. Next, AdaRPG leverages pretrained knowledge from GPT-4o to reason about manipulation mechanisms and generate control codes that invoke primitive skill functions for execution. By integrating foundation model reasoning, AdaRPG can automatically generalize to unseen object categories and effectively execute long-horizon adaptive policies.

To validate our proposed framework, we conduct experiments on seven categories of simulated objects and four categories of real-world objects, each exhibiting adaptive mechanisms. Through both simulation and real-world evaluations, we demonstrate that AdaRPG significantly improves cross-category generalization, outperforming existing methods in handling novel articulated objects.

In summary, our contributions include:

- We introduce a foundation model-powered framework that enhances generalization via part segmentation and mechanism reasoning for control code generation.
- We annotate a part-level affordance dataset to support generalizable affordance model guiding primitive skills.
- Extensive simulation and real-world experiments demonstrate the effectiveness of our AdaRPG framework.

2. Related Work

2.1. Adaptive Articulated Object Manipulation

Constructing simulation datasets for articulated object manipulation has been extensively studied [5, 14, 26, 30, 37, 46, 52]. Most previous works are restricted to simple, non-adaptive manipulation mechanisms. In contrast, AdaManip [49] introduces five distinct adaptive mechanisms in simulation, covering a broader range of real-world articulated objects. Given its comprehensive adaptive capabilities, we adopt AdaManip as the primary testing environment for our framework.

Alongside simulation environments, extensive research has been conducted on articulated object manipulation algorithms, spanning affordance-based [28, 38, 39, 48, 50, 51], flow-based [9, 25, 55], part-based [12, 13], RL-based [15] and diffusion-based [6, 49, 53] methods. For adaptive manipulation, AdaManip [49] collects adaptive demonstration to train an in-category adaptive diffusion policy. SAGE [13] leverages VLM and GAPartNet [14] to map semantic parts to actionable parts for executing manipulation policies with interactive adjustment.

2.2. Foundation Models for Robotics

With the remarkable generalization ability of foundation language and vision models, numerous studies have explored their integration into robotics.

One line of research focuses on mimicking the success of LLM by collecting massive datasets [4, 11, 16, 40, 47] to train large vision-language-action (VLA) models[3, 8, 22, 24, 29, 31, 45]. Another approach leverages existing foundation models for reward function specification in reinforcement learning[2, 7, 33–35, 54].

A different branch of research explores hierarchical policies, where high-level reasoning is performed by vision-

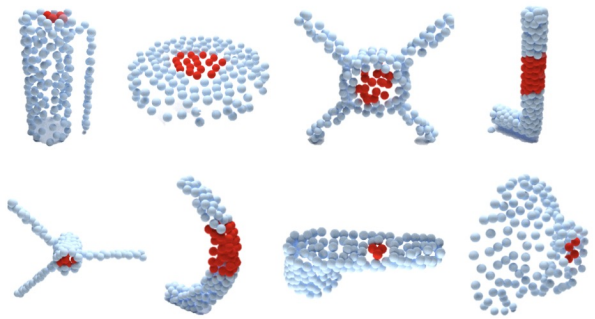

Figure 2. Visualization of several representative parts and affordance annotations in our Part Affordance Dataset.

language models (VLMs) that invoke low-level control functions to generate manipulation trajectories [1, 18, 19, 44]. Code as Policies[27] applies LLMs to generate robot control code, while VoxPoser[20] produces robot trajectories by generating value maps based on foundation models.

Recent studies SAGE[13] and CoPa[17] adopt object parts as an intermediate representation to constrain the policy, aligning with our approach. However, our method introduces a distinct part grounding pipeline and a part-affordance model to enable primitive skill execution, differentiating it from prior work.

3. Method

In this section, we introduce the three key components of AdaRPG, as illustrated in Fig. 1.

3.1. Part Dataset and Affordance Learning

As discussed in Sec. 1 and illustrated in Fig. 1, parts serve as a powerful intermediate representation for achieving significant generalization across object categories. Therefore, we aim to train a part affordance model that captures local geometric features for downstream execution.

Existing methods and datasets [38, 50] primarily focus on whole-object affordances. To address this limitation, we first construct and annotate a dataset with point-level affordance on parts. PartNet-Mobility [37, 52] is a comprehensive dataset of articulated objects with diverse geometries and categories. We select 11 categories of object from PartNet-Mobility and extract key functional parts, such as handles, buttons, and knobs. To ensure broad coverage, we carefully collect parts with diverse shapes and functionalities. The dataset statistics are summarized in Tab. 1. Note that our dataset only contains the **part point clouds** from PartNet-Mobility, without the full object context. The statistics in Tab. 1 are provided solely to illustrate the source object categories from which these parts are extracted. Therefore, no complete shapes have been included during training.

Given a part point cloud, we use an automated algorithm to annotate affordances on its surface. Human annotation is only required to specify the part's bounding box and affordance surface plane, based on the part's function characteristics. For instance, for a bottle cap, the annotation focuses on the top surface while ignoring the sides. Once the relevant surface is identified, high-affordance points are automatically computed based on the part's center and bounding box, completing the annotation. These high-affordance regions (highlighted in red) are centered around the part's center, representing areas associated with higher manipulation success rates. The part center is determined through a two-stage process, involving centroid estimation from the point cloud and subsequent refinement using the bounding box center. As the affordance surface defines the orientation plane for the gripper, its points can be directly used as end-effector targets without additional transformation.

The resulting dataset with point-level affordance on parts is represented as $D_{pa} = \{(O_i, p_i, r_i)\}$, where O_i is the part point cloud, p_i is the selected point, r_i is the actionability score, where $r_i = 1$ (positive) or $r_i = 0$ (negative) denote the annotated interaction outcome. We visualize several representative parts and corresponding affordance annotations in Fig. 2.

Following where2act [38] and VAT-MART [50], we employ Pointnet++ [41] as our affordance model encoder backbone, extracting per-point dense feature over object parts, followed by a MLP and a Sigmoid function mapping the output to actionability score $V(O_i, p_i) \in [0, 1]$. The training loss of affordance learning is the Binary Cross Entropy Loss:

$$L_V = \text{BCELoss}_{D_{pa}}(r_i, V(O_i, p_i)) \tag{1}$$

The affordance model V is trained on our proposed dataset, which includes diverse geometric and functional parts. As a result, it can predict the affordance of extracted parts from novel objects during online inference, provided they are correctly grounded and segmented. In the next section, we will elaborate on the process of grounding and segmenting these parts, ensuring accurate affordance prediction for previously unseen objects.

3.2. Part Grounding & Segmentation

Since part grounding and segmentation are essential prerequisites for affordance inference, ensuring their accuracy across all potential objects during online interaction is crucial. To enhance their performance, we incorporate foundation vision and language models, which are pretrained on extensive, diverse datasets. This large-scale pretraining equips them with strong generalization capabilities, enabling them to recognize and interpret a wide range of object parts and features more effectively.

As illustrated in Fig. 3, the camera captures RGB images and sends them to GPT-4o [21]. We prompt GPT-4o to iden-

Categories	Window	Bottle	CM	Door	Safe	Faucet	Pot	Microwave	Pen	Switch	Toaster
Part Count	4	14	4	20	20	14	15	7	11	7	8

Table 1. **Statistics** of our Part Affordance Dataset. Note that our dataset only contains the **part point clouds** derived from PartNet-Mobility, without the full object context. The part counts indicate how many parts are extracted from each category. **CM** stands for coffee machine.

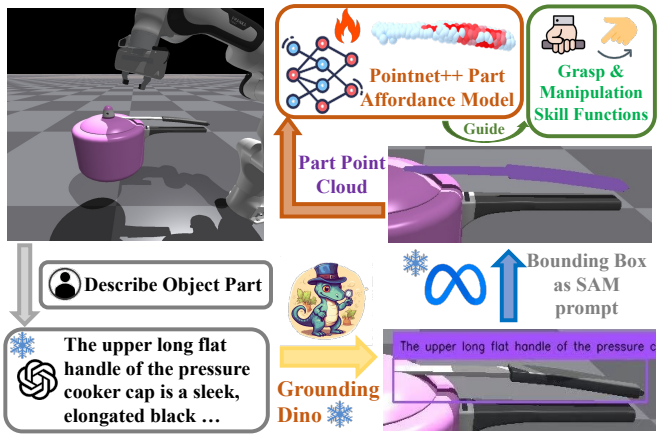

Figure 3. Foundation Models Guided Part Grounding and Segmentation. The RGB image is first processed by GPT-4o to generate a detailed part description, which is then used as an input prompt for GroundingDINO. GroundingDINO generates a bounding box for the identified part, which serves as the input prompt for Segment Anything to perform precise segmentation. The segmented part's point cloud is subsequently fed into the pre-trained part affordance model to infer an affordance score, guiding downstream primitive skills for manipulation.

tify the scene and object, then describe the functional part in no more than three sentences while ignoring the robot arm and carefully distinguishing different object parts. For example, in Fig. 3, the pressure cooker has two handles: the lower handle is fixed to the cooker body and is non-actionable, whereas the upper handle is movable and thus relevant for interaction.

GPT-4o generates a detailed part description, which is then used as an input prompt for the GroundingDINO [32, 43] model to generate a bounding box for the corresponding part. Notably, our extensive experimental trials confirm that providing a detailed description as a prompt significantly improves GroundingDINO's performance compared to using single-word prompts.

The bounding box generated by GroundingDINO serves as the input prompt for Segment Anything [23, 42], which refines the segmentation by precisely delineating the target part from the surrounding object structure. This process ensures that only the relevant actionable component is isolated while filtering out extraneous details. The obtained segmentation mask is then back-projected onto the depth image, enabling the reconstruction of an accurate 3D part representation in the form of a segmented point cloud. This step is crucial, as it allows for the preservation of spatial and geometric information, which is essential for downstream affordance prediction.

The segmented part point cloud is then processed by the pre-trained affordance model, which generates a dense actionability score map, guiding downstream primitive skills such as grasping and rotating.

As depicted in Fig. 3, all foundation models (GPT, GroundingDINO, and SAM) operate in a frozen, training-free manner, requiring no additional fine-tuning. Instead, they rely on extensive pretraining on large-scale datasets to provide high-quality part identification, localization, and segmentation. The only trainable component in our framework is the affordance model, which is specifically optimized on our proposed dataset. This strategic combination of powerful pre-trained models with a task-specific affordance predictor enables our approach to achieve both broad generalization and task-adaptive precision.

3.3. Primitive Skill as Function

Once we obtain the affordance score of the part point cloud, we identify actionable interaction points by selecting those with scores above a predefined threshold ϵ. The end-effector's translation pose is then computed as the average position of these selected points. To determine the gripper orientation, we estimate the normal direction of the interaction point on the point cloud surface, ensuring an optimal grasp pose. This process forms the primitive grasp function, which serves as the initial step for further manipulation.

Beyond grasping, we implement six atomic manipulation functions centered on the robot's end-effector, defined along two principal axes of the end-effector coordinate frame: the z-axis (aligned with the gripper's pointing direction) and the y-axis (parallel to the gripper's two fingers and perpendicular to the z-axis). For z-axis, two translation actions (push/pull) move the end-effector along its pointing direction, while two self rotational actions allow the gripper to rotate around its own axis. For y-axis, the two translation actions refer to moving the part around the object axis. Each atomic action function returns a success flag upon completion, serving as feedback from the environment. For example, the *pull_part* function returns success when the end-effector's movement reaches a predefined step size.

To ensure smooth and stable execution, all movements are governed by impedance control, which dynamically ad-

justs force and compliance to enhance interaction stability. By defining atomic actions in the end-effector's local frame, our approach enables flexible and adaptive manipulation of objects in diverse orientations. The translation-based actions move the end-effector by a fixed step in the specified direction, while the rotational actions execute controlled rotations. To maintain precision, we continuously adjust the end-effector by computing and correcting angle deviations between its actual and expected positions in real-time. This correction mechanism, combined with impedance control, improves robustness across diverse object geometries and interaction scenarios, making the primitive skills more robust functions for high-level programs.

3.4. High-Level Code Generation

With the primitive skill functions, we leverage foundation model reasoning to generate high-level Python control code. Using natural language input, GPT-4o generates structured code that dynamically orchestrates primitive skills such as grasping, pulling, and rotating to adaptively manipulate articulated objects. All code is produced from a single general prompt and follows a unified generation process, ensuring scalability across object categories.

The generated code example follows an adaptive control loop, as shown in Fig. 4. The process begins with a grasp action, followed by repeated rotation attempts to adjust the part's position. A probability-based condition, which increases with the number of rotations, determines whether a pull attempt is made. If successful, the part is considered unlocked, and the process transitions to a secondary loop that continues executing the *pull_part* function until the task is complete. This approach ensures that the robot persistently adapts to unexpected constraints, increasing robustness in manipulation.

By leveraging foundation models to generate structured control logic, our method enhances generalization across different articulated objects, eliminating the need for manually scripted task-specific policies. The resulting code can be seamlessly integrated into real-world robotic systems, demonstrating the effectiveness of our framework in mechanism reasoning and high-level control strategies.

4. Experiments

4.1. Simulation Settings

We perform cross-category generalization experiments using the AdaManip simulation environment[49], which is built on IsaacGym[36]. Our evaluation spans seven distinct object categories: bottle, pen, pressure cooker, coffee machine, window, door, and lamp, covering a wide range of articulated mechanisms and functional variations.

A key aspect of our evaluation is that our affordance model is not trained on the AdaManip dataset, and our part

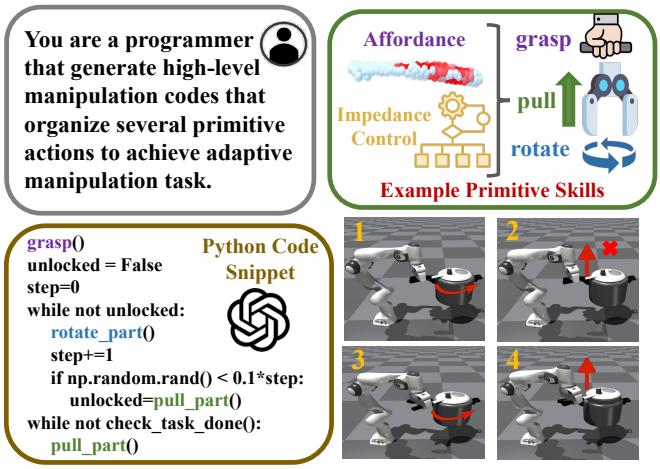

Figure 4. Foundation Model Reasoning and High-Level Python Control Code Generation. The lower-left image displays a Python snippet example generated by GPT-4o, which orchestrates adaptive manipulation by calling the grasp, pull, and rotate part functions. The code forms a while loop, ensuring the robot continues rotating if the pulling action fails, enabling iterative refinement for successful manipulation.

affordance training set derived from PartNet-Mobility lacks object-level annotations. This ensures that our framework is tested on entirely **novel object categories**, making it an unbiased assessment of its generalization capability.

To quantitatively evaluate performance, we report IoU between predicted and ground-truth part segmentation masks, F1 score for part affordance prediction, and manipulation success rate across all object categories. These metrics assess both individual components of our framework and the overall policy's ability to generalize to previously unseen articulated objects. Achieving a high success rate requires not only accurate grounding of part segmentation and affordance, but also the ability to generate structured control code that reliably sequences primitive skills to complete tasks in a stable and efficient manner. Together, these evaluations highlight the framework's capacity for complex, adaptive interaction with diverse articulated objects.

4.2. Baselines and Ablations

To thoroughly evaluate the effectiveness of our proposed method in handling complex adaptive manipulation tasks, we conduct a comprehensive comparison with state-of-the-art approaches leveraging large language models (LLMs) and imitation learning (IL) relying on expert adaptive demonstrations to infer manipulation policies. Additionally, we include two ablated versions of our method to evaluate the contribution of individual components.

While LLM-based methods excel in semantic reasoning and task decomposition, they often struggle with precise execution due to limited physical understanding and lack

of real-world affordance grounding. In contrast, IL-based methods are proficient in replicating expert behaviors, but their performance is constrained by dataset diversity and difficulty in adapting to novel objects.

- **VAT-MART**[50], an affordance-based method that predicts open-loop manipulation actions. Here we use VAT-MART solely as a baseline for affordance prediction.
- **SAGE**[13], maps semantic parts of articulated objects to Generalizable Actionable Parts (GAParts) using a vision-language model and a domain-specific model GAPartNet[14] to generate executable manipulation policies from natural language instructions with interactive feedback for robustness.
- **CoPa**[17], a robotic manipulation framework that leverages foundation vision-language models to generate 6-DoF end-effector poses by integrating task-oriented grasping and task-aware motion planning. It selects grasping parts using a coarse-to-fine grounding mechanism and derives post-grasp poses by identifying spatial geometry constraints of task-relevant object parts.
- **AdaManip**[49], an imitation learning approach that trains a diffusion policy using rule-based adaptive demonstration data. Since the original AdaManip algorithm is designed for in-category generalization, we extend its training to include all seven object categories. However, the test objects remain entirely novel to the training set.
- **Ours w/o prompt**. Part segmentation masks are generated without GPT-4o descriptions, using a generic prompt for GroundingDINO instead.
- **Ours w/o affordance**. The affordance module is removed, and manipulation points are set to the center of the segmented part, effectively treating all regions as equally actionable (i.e., assigning uniform affordance scores).

4.3. Simulation Results

4.3.1. Part Segmentation

We begin by evaluating the robustness of the foundation models in part grounding and segmentation, which constitutes the first stage of our inference pipeline. We use Intersection over Union (IoU) as the evaluation metric, comparing the predicted part segmentation masks, generated by our pipeline (GPT-4o + GroundingDINO + SAM), against ground-truth masks. As shown in Table 2, the IoU exceeds 80% across all object categories, demonstrating reliable performance in part grounding and segmentation.

Categories	Bottle	Pen	PC	CM	Window	Door	Lamp
IoU	0.99	0.97	0.84	0.87	0.88	0.99	0.80

Table 2. **IoU between predicted and GT part masks.** PC and CM stand for Pressure Cooker and Coffee Machine, respectively. The same abbreviations are used in the following tables.

4.3.2. Affordance Prediction

We further evaluate the performance of our affordance prediction module, benchmarking against VAT-MART [50], an affordance-based method for articulated object manipulation. Both models are trained on affordance dataset derived from PartNet-Mobility. While VAT-MART learns to predict affordance scores over the entire object point cloud, our model takes as input only the part point cloud.

During inference, VAT-MART predicts affordance scores for the full object, which we then mask using the ground-truth part segmentation to isolate the part-level prediction. In our approach, we directly extract the ground-truth part point cloud and predict affordance scores at the part level. Consequently, all evaluations are conducted on part-level affordance predictions. This allows us to ablate the affordance prediction performance from the influence of upstream part segmentation.

To assess cross-category generalization, we evaluate both methods on seven object categories from the AdaManip dataset using the F1-score metric. As shown in Table 3, our model significantly outperforms VAT-MART. We attribute this to VAT-MART's reliance on whole-object geometry, which introduces high geometric variability across categories. In contrast, our part-level formulation reduces geometric variance, enabling more robust and generalizable affordance prediction.

Method	Bottle	Pen	PC	CM	Window	Door	Lamp
VAT-MART[50]	0.56	0.18	0.21	0.29	0.34	0.17	0.16
Ours	**0.82**	**0.66**	**0.72**	**0.83**	**0.91**	**0.61**	**0.64**

Table 3. **Part Affordance F1 score comparison.**

4.3.3. Object Manipulation

We compare AdaRPG with baselines and ablations on the final object manipulation success rates. We present the quantitative results in Tab. 4, which demonstrate that AdaRPG outperforms all baselines and ablations across all object categories. Notably, while AdaManip is trained on a larger dataset covering all seven categories, SAGE, CoPa, and AdaRPG remain entirely unexposed to the test data, making AdaRPG's superior generalization even more compelling.

Despite being VLM-powered zero-shot manipulation frameworks, both SAGE and CoPa fall behind AdaRPG due to key limitations in their part representation and execution strategies. SAGE relies on GAPartNet for part grounding, which processes the entire point cloud and predicts part poses for rule-based primitive actions. However, pose estimation lacks the precision of affordance-based representations, leading to unstable execution as errors in part type estimation propagate through the manipulation pipeline.

On the other hand, CoPa incorporates GraspNet [10], a foundation grasp model trained on large-scale grasping

Task	Adaptive Manipulation						
Method	**Bottle**	**Pen**	**PC**	**CM**	**Window**	**Door**	**Lamp**
SAGE[13]	0.21	0.40	0.00	0.30	0.38	0.39	0.40
CoPa[17]	0.58	0.47	0.17	0.40	0.08	0.39	0.20
AdaManip[49]	0.46	0.53	0.50	0.60	0.46	0.44	0.30
Ours w/o prompt	0.11	0.00	0.00	0.00	0.00	0.00	0.20
Ours w/o affordance	0.78	0.58	0.83	0.60	0.57	0.63	0.40
Ours	**0.84**	**0.73**	**1.00**	**0.80**	**0.84**	**0.78**	**0.70**

Table 4. **Object Manipulation Success Rates** of baselines, ablations and our method in simulation. Our proposed AdaRPG outperforms baseline and ablation methods in all categories.

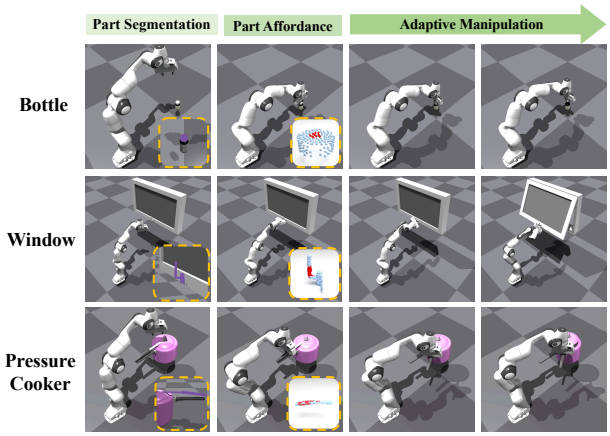

Figure 5. **Qualitative Simulation Result.** The intermediate results of part segmentation and affordance prediction demonstrate the effectiveness of AdaRPG.

data. While GraspNet excels at generic object grasping, our tasks require functional grasping, where the robot must interact with specific parts to complete sequential manipulation tasks. In practice, most grasp candidates generated by GraspNet fail to align with the actionable parts of articulated objects, leading to failures early in execution.

In contrast, AdaRPG utilizes a part affordance model to generate a fine-grained actionability score, enabling precise impedance-control-based primitive skills. By integrating multiple foundation models for part grounding and reasoning, AdaRPG achieves significantly higher success rates, demonstrating superior generalization and stability across diverse articulated objects.

To further analyze our method, we perform an ablation study. As shown in Tab. 4, removing prompt guidance for part segmentation results in failures due to inaccurate part grounding. The affordance module significantly improves contact point selection, leading to a 15% higher average success rate compared to the w/o-affordance baseline, highlighting its strong generalization across parts.

Additional qualitative results are visualized in Fig. 5, visualizing AdaRPG's performance across three selected object categories. The results demonstrate that foundation-guided part grounding and affordance-based generalization effectively identify actionable parts, ensuring stable and precise manipulation. This strong part segmentation and affordance reasoning provide a robust foundation for high-performance execution, enabling AdaRPG to generalize well to novel articulated objects.

4.4. Real-World Experiments

Method	Bottle	Pressure Cooker	Microwave	Lamp
SAGE[13]	5/10	6/10	3/10	6/10
CoPa[17]	4/10	2/10	3/10	1/10
AdaManip[49]	8/10	5/10	7/10	5/10
Ours	**9/10**	**10/10**	**9/10**	**8/10**

Table 5. **Real-World Object Manipulation Success Rates.**

To evaluate the real-world generalization of our framework, we conduct experiments on a diverse set of everyday articulated objects, including a pressure cooker, microwave, bottle, and lamp. For these experiments, we utilize a Franka Emika Panda robotic arm equipped with an Intel RealSense D415 depth camera.

Following our framework, we first extract the part mask from the RGB image using foundation models, ensuring accurate part segmentation. The segmented part point cloud is then processed to predict per-point affordance scores, identifying the most actionable regions for interaction. The point with the highest affordance score is selected as the optimal grasping location, after which the robot executes primitive manipulation skills such as pulling, rotating, or pushing to complete the task. This pipeline enables adaptive and precise real-world interaction, demonstrating the robustness of our affordance-based approach in handling diverse articulated objects.

The quantitative results of real-world experiments are presented in Tab. 5, showing that all three baselines underperform compared to AdaRPG. Notably, by comparing Tab. 5 with Tab. 4, we observe that the real-world performance of AdaRPG surpasses simulation results. This improvement can be attributed to the fact that all foundation models are trained on real-world data, whereas the simulation environment introduces a domain gap that slightly hinders performance. This finding suggests that incorporating more foundation models into a robotic system could further enhance its ability to handle real-world challenges, leveraging their strong generalization capabilities to bridge perception and control effectively.

We present qualitative results from real-world experiments in Fig. 6, demonstrating the effectiveness of our approach in part recognition and segmentation across all four object categories. The results confirm that our foundation-guided part grounding pipeline successfully identifies and isolates actionable components, enabling precise manipula-

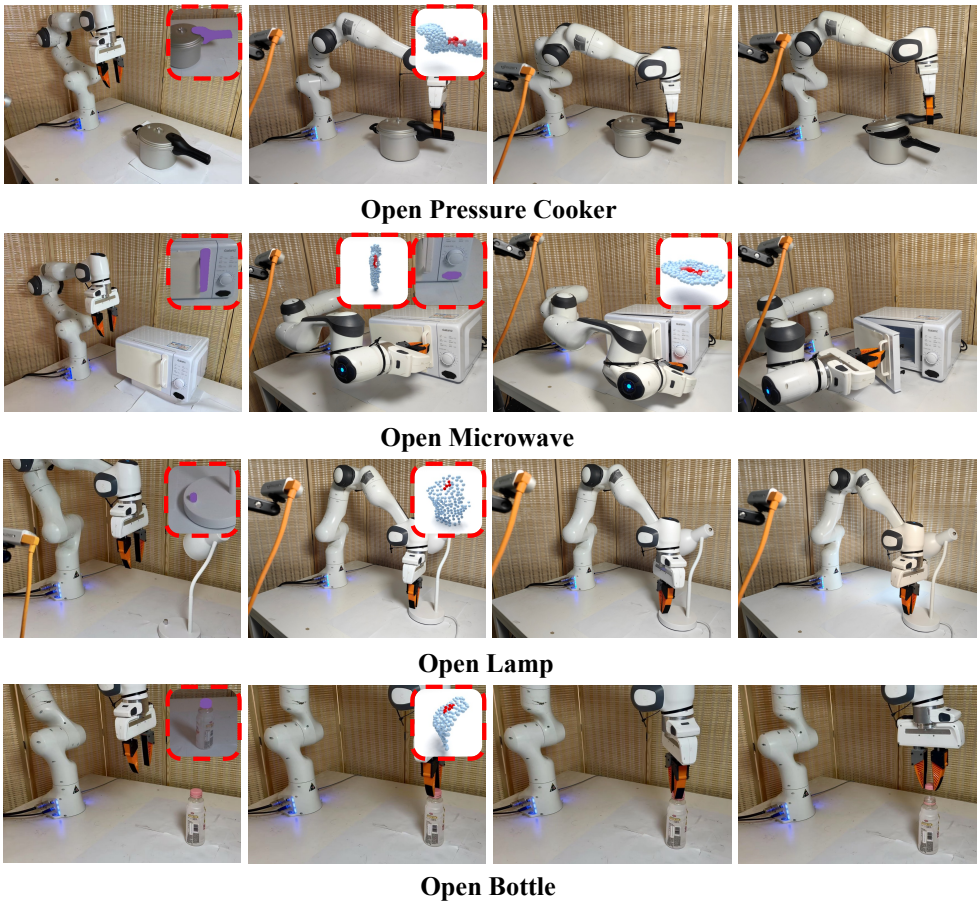

Figure 6. **Qualitative Real-World Results**. We visualize the segmented handle, knob, and cap parts, along with their corresponding affordance score predictions. The high-level control codes leverage affordance-guided primitive actions to successfully complete the tasks.

tion. Additionally, the affordance model trained on simulation data transfers effectively to real-world point clouds, reinforcing the robustness and adaptability of affordance-based representations. This successful transfer further validates our design choice, highlighting the advantages of part-level affordance modeling over alternative representation strategies in generalizing across diverse environments.

5. Conclusion

In this work, we presented **AdaRPG**, a novel framework that leverages foundation models to enhance the generalization of adaptive articulated object manipulation by segmenting object parts and reasoning about their functional affordances. By shifting the focus from whole-object to part-level affordance inference, AdaRPG overcomes the challenges posed by geometric diversity and functional variability in real-world articulated objects. Our approach integrates a foundation model-powered pipeline with a structured affordance model, enabling adaptive primitive skill execution through high-level control code generation.

To support this framework, we introduced a part-level affordance annotation dataset, which provides a structured learning foundation for affordance inference, guiding precise manipulation strategies across diverse object categories. Experimental results in both simulation and real-world environments validate the robust generalization capability of AdaRPG, demonstrating its effectiveness in handling novel articulated objects beyond training data.

Moving forward, we aim to further enhance AdaRPG's adaptability by integrating interactive affordance learning under dynamic environmental conditions. We hope that this work can pave the way for more generalizable and adaptable robotic manipulation strategies, bridging the gap between vision-language models and real-world robotic control.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grants 62176025, 62301066, U21B2045, and 62206012.

References

[1] Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, et al. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691*, 2022. 3

[2] Minttu Alakuijala, Gabriel Dulac-Arnold, Julien Mairal, Jean Ponce, and Cordelia Schmid. Learning reward functions for robotic manipulation by observing humans. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5006–5012. IEEE, 2023. 2

[3] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. pi_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024. 2

[4] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022. 2

[5] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, L. Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository. *ArXiv*, abs/1512.03012, 2015. 2

[6] Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric A. Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *ArXiv*, abs/2303.04137, 2023. 2

[7] Yuchen Cui, Scott Niekum, Abhinav Gupta, Vikash Kumar, and Aravind Rajeswaran. Can foundation models perform zero-shot task specification for robot manipulation? In *Learning for dynamics and control conference*, pages 893–905. PMLR, 2022. 2

[8] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03778*, 2023. 2

[9] Ben Eisner, Harry Zhang, and David Held. Flowbot3d: Learning 3d articulation flow to manipulate articulated objects. *arXiv preprint arXiv:2205.04382*, 2022. 2

[10] Hao-Shu Fang, Chenxi Wang, Minghao Gou, and Cewu Lu. Graspnet-1billion: A large-scale benchmark for general object grasping. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11444–11453, 2020. 6

[11] Hao-Shu Fang, Hongjie Fang, Zhenyu Tang, Jirong Liu, Chenxi Wang, Junbo Wang, Haoyi Zhu, and Cewu Lu. Rh20t: A comprehensive robotic dataset for learning diverse skills in one-shot. *arXiv preprint arXiv:2307.00595*, 2023. 2

[12] Haoran Geng, Ziming Li, Yiran Geng, Jiayi Chen, Hao Dong, and He Wang. Partmanip: Learning cross-category generalizable part manipulation policy from point cloud observations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2978–2988, 2023. 2

[13] Haoran Geng, Songlin Wei, Congyue Deng, Bokui Shen, He Wang, and Leonidas Guibas. Sage: Bridging semantic and actionable parts for generalizable articulated-object manipulation under language instructions, 2023. 2, 3, 6, 7

[14] Haoran Geng, Helin Xu, Chengyang Zhao, Chao Xu, Li Yi, Siyuan Huang, and He Wang. 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*, pages 7081–7091, 2023. 2, 6

[15] Yiran Geng, Boshi An, Haoran Geng, Yuanpei Chen, Yaodong Yang, and Hao Dong. End-to-end affordance learning for robotic manipulation. *ICRA*, 2023. 2

[16] Ran Gong, Jiangyong Huang, Yizhou Zhao, Haoran Geng, Xiaofeng Gao, Qingyang Wu, Wensi Ai, Ziheng Zhou, Demetri Terzopoulos, Song-Chun Zhu, et al. Arnold: A benchmark for language-grounded task learning with continuous states in realistic 3d scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 2

[17] Haoxu Huang, Fanqi Lin, Yingdong Hu, Shengjie Wang, and Yang Gao. Copa: General robotic manipulation through spatial constraints of parts with foundation models. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 9488–9495. IEEE, 2024. 3, 6, 7

[18] Wenlong Huang, Pieter Abbeel, Deepak Pathak, and Igor Mordatch. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. In *International conference on machine learning*, pages 9118–9147. PMLR, 2022. 3

[19] Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, et al. Inner monologue: Embodied reasoning through planning with language models. *arXiv preprint arXiv:2207.05608*, 2022. 3

[20] Wenlong Huang, Chen Wang, Ruohan Zhang, Yunzhu Li, Jiajun Wu, and Li Fei-Fei. Voxposer: Composable 3d value maps for robotic manipulation with language models. *arXiv preprint arXiv:2307.05973*, 2023. 3

[21] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 3

[22] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024. 2

[23] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023. 4

[24] Qixiu Li, Yaobo Liang, Zeyu Wang, Lin Luo, Xi Chen, Mozheng Liao, Fangyun Wei, Yu Deng, Sicheng Xu,

Yizhong Zhang, et al. Cogact: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation. *arXiv preprint arXiv:2411.19650*, 2024. 2

[25] Yishu Li, Wen Hui Leng, Yiming Fang, Ben Eisner, and David Held. Flowbothd: History-aware diffuser handling ambiguities in articulated objects manipulation. *arXiv preprint arXiv:2410.07078*, 2024. 2

[26] Yu Li, Xiaojie Zhang, Ruihai Wu, Zilong Zhang, Yiran Geng, Hao Dong, and Zhaofeng He. Unidoormanip: Learning universal door manipulation policy over large-scale and diverse door manipulation environments. *arXiv preprint arXiv:2403.02604*, 2024. 2

[27] Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. Code as policies: Language model programs for embodied control. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9493–9500. IEEE, 2023. 3

[28] Suhan Ling, Yian Wang, Shiguang Wu, Yuzheng Zhuang, Tianyi Xu, Yu Li, Chang Liu, and Hao Dong. Articulated object manipulation with coarse-to-fine affordance for mitigating the effect of point cloud noise. *ICRA*, 2024. 2

[29] Jiaming Liu, Mengzhen Liu, Zhenyu Wang, Pengju An, Xiaoqi Li, Kaichen Zhou, Senqiao Yang, Renrui Zhang, Yandong Guo, and Shanghang Zhang. Robomamba: Efficient vision-language-action model for robotic reasoning and manipulation. *Advances in Neural Information Processing Systems*, 37:40085–40110, 2024. 2

[30] Liu Liu, Wenqiang Xu, Haoyuan Fu, Sucheng Qian, Qiaojun Yu, Yang Han, and Cewu Lu. Akb-48: A real-world articulated object knowledge base. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14809–14818, 2022. 2

[31] Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. Rdt-1b: a diffusion foundation model for bimanual manipulation. *arXiv preprint arXiv:2410.07864*, 2024. 2

[32] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European Conference on Computer Vision*, pages 38–55. Springer, 2024. 4

[33] Yecheng Jason Ma, Shagun Sodhani, Dinesh Jayaraman, Osbert Bastani, Vikash Kumar, and Amy Zhang. Vip: Towards universal visual reward and representation via value-implicit pre-training. *arXiv preprint arXiv:2210.00030*, 2022. 2

[34] Yecheng Jason Ma, William Liang, Guanzhi Wang, De-An Huang, Osbert Bastani, Dinesh Jayaraman, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Eureka: Human-level reward design via coding large language models. *arXiv preprint arXiv:2310.12931*, 2023.

[35] Parsa Mahmoudieh, Deepak Pathak, and Trevor Darrell. Zero-shot reward specification via grounded natural language. In *International Conference on Machine Learning*, pages 14743–14752. PMLR, 2022. 2

[36] Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, et al. Isaac gym: High performance gpu-based physics simulation for robot learning. *arXiv preprint arXiv:2108.10470*, 2021. 5

[37] Kaichun Mo, Shilin Zhu, Angel X. Chang, Li Yi, Subarna Tripathi, Leonidas J. Guibas, and Hao Su. PartNet: A large-scale benchmark for fine-grained and hierarchical part-level 3D object understanding. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 3

[38] Kaichun Mo, Leonidas J. Guibas, Mustafa Mukadam, Abhinav Gupta, and Shubham Tulsiani. Where2act: From pixels to actions for articulated 3d objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6813–6823, 2021. 2, 3

[39] Chuanruo Ning, Ruihai Wu, Haoran Lu, Kaichun Mo, and Hao Dong. Where2explore: Few-shot affordance learning for unseen novel categories of articulated objects. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 2

[40] Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6892–6903. IEEE, 2024. 2

[41] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 3

[42] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 4

[43] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024. 4

[44] Ishika Singh, Valts Blukis, Arsalan Mousavian, Ankit Goyal, Danfei Xu, Jonathan Tremblay, Dieter Fox, Jesse Thomason, and Animesh Garg. Progprompt: Generating situated robot task plans using large language models. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11523–11530. IEEE, 2023. 3

[45] Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, et al. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*, 2024. 2

[46] Yusuke Urakami, Alec Hodgkinson, Casey Carlin, Randall Leu, Luca Rigazio, and Pieter Abbeel. Doorgym: A scalable door opening environment and baseline agent. *arXiv preprint arXiv:1908.01887*, 2019. 2

[47] Homer Rich Walke, Kevin Black, Tony Z Zhao, Quan Vuong, Chongyi Zheng, Philippe Hansen-Estruch, Andre Wang He, Vivek Myers, Moo Jin Kim, Max Du, et al.

Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning*, pages 1723–1736. PMLR, 2023. 2

[48] Yian Wang, Ruihai Wu, Kaichun Mo, Jiaqi Ke, Qingnan Fan, Leonidas Guibas, and Hao Dong. Adaafford: Learning to adapt manipulation affordance for 3d articulated objects via few-shot interactions. *European conference on computer vision (ECCV 2022)*, 2022. 2

[49] Yuanfei Wang, Xiaojie Zhang, Ruihai Wu, Yu Li, Yan Shen, Mingdong Wu, Zhaofeng He, Yizhou Wang, and Hao Dong. Adamanip: Adaptive articulated object manipulation environments and policy learning. In *International Conference on Learning Representations*, 2025. 2, 5, 6, 7

[50] Ruihai Wu, Yan Zhao, Kaichun Mo, Zizheng Guo, Yian Wang, Tianhao Wu, Qingnan Fan, Xuelin Chen, Leonidas Guibas, and Hao Dong. VAT-mart: Learning visual action trajectory proposals for manipulating 3d ARTiculated objects. In *International Conference on Learning Representations*, 2022. 2, 3, 6

[51] Ruihai Wu, Kai Cheng, Yan Zhao, Chuanruo Ning, Guanqi Zhan, and Hao Dong. Learning environment-aware affordance for 3d articulated object manipulation under occlusions. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 2

[52] Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao Jiang, Yifu Yuan, He Wang, et al. Sapien: A simulated part-based interactive environment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11097–11107, 2020. 2, 3

[53] Yanjie Ze, Gu Zhang, Kangning Zhang, Chenyuan Hu, Muhan Wang, and Huazhe Xu. 3d diffusion policy. *ArXiv*, abs/2403.03954, 2024. 2

[54] Yuwei Zeng, Yao Mu, and Lin Shao. Learning reward for robot skills using large language models via self-alignment. *arXiv preprint arXiv:2405.07162*, 2024. 2

[55] Harry Zhang, Ben Eisner, and David Held. Flowbot++: Learning generalized articulated objects manipulation via articulation projection. *arXiv preprint arXiv:2306.12893*, 2023. 2