

000 001 002 003 004 005 006 007 008 009 010 011 012 013 014 015 016 017 018 019 020 021 022 023 024 025 026 027 028 029 030 031 032 033 034 035 036 037 038 039 040 041 042 043 044 045 046 047 048 049 050 051 052 053 LEARNING PART-AWARE DENSE 3D FEATURE FIELD FOR GENERALIZABLE ARTICULATED OBJECT MANIP- ULATION

Anonymous authors

Paper under double-blind review

ABSTRACT

Articulated object manipulation is essential for various real-world robotic tasks, yet generalizing across diverse objects remains a major challenge. A key to generalization lies in understanding functional parts (e.g., door handles and knobs), which indicate where and how to manipulate across diverse object categories and shapes. Previous works attempted to achieve generalization by introducing foundation features, while these features are mostly 2D-based and do not specifically consider functional parts. When lifting these 2D features to geometry-profound 3D space, challenges arise, such as long runtimes, multi-view inconsistencies, and low spatial resolution with insufficient geometric information. To address these issues, we propose **Part-Aware 3D Feature Field (PA3FF)**, a novel dense 3D feature with part awareness for generalizable articulated object manipulation. PA3FF is trained by 3D part proposals from a large-scale labeled dataset, via a contrastive learning formulation. Given point clouds as input, PA3FF predicts a continuous 3D feature field in a feedforward manner, where the distance between point features reflects the proximity of functional parts: points with similar features are more likely to belong to the same part. Building on this feature, we introduce the **Part-Aware Diffusion Policy (PADP)**, an imitation learning framework aimed at enhancing sample efficiency and generalization for robotic manipulation. We evaluate PADP on several simulated and real-world tasks, demonstrating that PA3FF consistently outperforms a range of 2D and 3D representations in manipulation scenarios, including CLIP, DINOv2, and Grounded-SAM, achieving state-of-the-art performance. Beyond imitation learning, PA3FF enables diverse downstream methods, including correspondence learning and segmentation tasks, making it a versatile foundation for robotic manipulation. Project page: <https://pa3ff.github.io/>.

1 INTRODUCTION

The next generation of assistive robots must possess the generalization ability to manipulate objects across a broad range of scenarios with ease and adaptability. To achieve this goal, recent studies (Black et al., 2024; Intelligence et al., 2025; Team et al., 2024; Kim et al., 2024b) leverage the power of 2D vision-language foundation models (e.g., CLIP (Radford et al., 2021), DINOv2 (Oquab et al., 2023), SigLIP (Zhai et al., 2023)) to improve performance and generalization in robotic manipulation policies. However, these representations inherently lack 3D geometry and spatial continuity, which are crucial for reasoning about object shapes, part configurations, and affordance in manipulation tasks (Zhu et al., 2024; Zhang et al., 2023; Ke et al., 2024a).

Some recent works attempt to lift 2D features into 3D feature fields via multi-view fusion or neural rendering (Kerr et al., 2023; Shen et al., 2023; Lin et al., 2023; Rashid et al., 2023; Ze et al., 2023). While these methods improve the understanding of 3D objects and scenes, they are not native 3D representations, and thus usually suffer from problems such as **long inference times** (even minutes), **feature inconsistency across views** (Wang et al., 2024b), and **limited spatial resolution** (Fu et al., 2024), making them impractical for dense, fine-grained and real-time robotic manipulation.

In this paper, we propose **Part-Aware 3D Feature Field (PA3FF)**, a novel 3D-native representation designed to encode dense, semantic, and part-aware features directly from point clouds. PA3FF

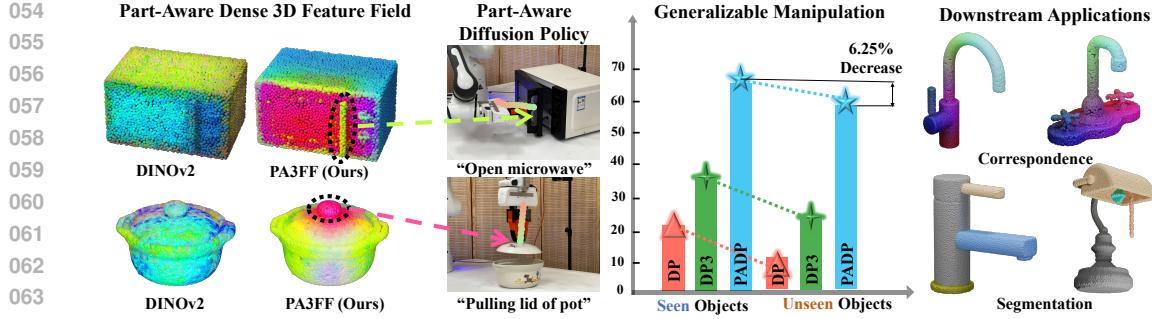


Figure 1: (1) We propose PA3FF, a feedforward model that predicts part-aware 3D feature fields for 3d shapes. (2) We propose a part-aware diffusion policy, which leverages PA3FF, that can efficiently generalize to unseen objects. (3) PA3FF exhibits consistency across shapes, enabling various downstream applications such as correspondence and segmentation.

predicts a continuous 3D feature field in a feedforward manner, where the distance between features reflects the part-level awareness—points with similar features are more likely to belong to the same part. We leverage Sonata, a model pretrained on 140k point clouds with self distillation, to provide rich 3D geometric priors for our proposed representation. To enhance the part-awareness, we incorporate a contrastive learning framework that establishes consistent relationships between 3D part-level features and their corresponding semantic counterparts by a collection of public dataset (PartNet-Mobility (Xiang et al., 2020; Mo et al., 2018), 3DCoMPaT (Slim et al., 2025), PartObjaverse-Tiny (Yang et al., 2024b)).

To demonstrate the power of PA3FF in robotic manipulation scenarios, we introduce **Part-Aware Diffusion Policy (PADP)**, a 3D point cloud-based visuomotor policy that integrates PA3FF with a diffusion policy architecture, as shown in Figure 1 (Left). The part awareness and generalization capability to novel shapes empower the highly sample-efficient and generalizable manipulation behaviors across objects. As shown in Figure 1 (Right), beyond imitation learning, PA3FF enables diverse downstream methods, including correspondence learning, along with key-point proposals for planning constraints, making it a versatile foundation for robotic manipulation.

We evaluate PA3FF and PADP on a broad spectrum of robotic manipulation tasks from the PartInstruct (Yin et al., 2025), as well as in the real world. PADP sets a new state-of-the-art on PartInstruct with a 9.4% absolute gain, outperforming existing 2D and 3D representations with Diffusion Policy (Chi et al., 2024a), such as CLIP (Radford et al., 2021), DINOv2 (Oquab et al., 2023) and Grounded-SAM (Ren et al., 2024). We further show PADP significantly surpasses a strong baseline (GenDP (Wang et al., 2024c)) from 8 real-world tasks, offering 18.75% increment. We also provide a detailed analysis of how well and why our method can generalize to novel instances.

In summary, our contributions include:

- We introduce **PA3FF**, a 3D-native representation that encodes dense, semantic, and functional part-aware features directly from point clouds.
- We develop **PADP**, a diffusion policy that leverages PA3FF for generalizable manipulation with strong sample efficiency.
- PA3FF can further enable diverse downstream methods, including correspondence learning and segmentation, making it a versatile foundation for robotic manipulation.
- We validate our approach on 16 PartInstruct and 8 real-world tasks, where it significantly outperforms prior 2D and 3D representations (CLIP, DINOv2, and Grounded-SAM), offering a 15% and 16.5% increase.

2 RELATED WORK

3D Semantic Representation in Robotic Manipulation. One common approach to 3D semantic representation involves extracting functional or affordance information from observations Paulius et al. (2016); Chen et al. (2022); Kokic et al. (2017); Wu & Zhao (2022); Zhao et al. (2023b); Wang et al. (2022); Wu et al. (2023); Wen et al. (2022); Di Palo & Johns (2024). This line of research

108 focuses on tasks that can be accomplished using motion primitives such as grasping, picking, and
 109 placing. However, our diffusion-based policy offers greater flexibility in terms of action representation
 110 and task execution. Another approach lifts 2D foundational features (e.g., CLIP, DINOv2) to 3D
 111 representations via multi-view fusion (Wang et al., 2024b; Kerr et al., 2023; Wang et al., 2024c; Ke
 112 et al., 2024a; Yang et al., 2024b). However, this method has several notable limitations. First, multi-
 113 view fusion in previous approaches is computationally expensive and suffers from inconsistent features
 114 across views. Second, these methods often sacrifice spatial resolution for semantic quality in 2D
 115 features. For instance, ViT-family models process image tokens as patches, resulting in significantly
 116 lower-resolution feature maps (e.g., 14x smaller in DINOv2), which leads to a substantial loss of
 117 spatial information. In contrast, our approach leverages visual representations that are pre-trained on
 118 large-scale point clouds, which enables our robot to generalize to unseen configurations efficiently.

119 **Imitation Learning.** Imitation learning (IL) has proven effective in enabling end-to-end robotic
 120 training from expert demonstrations via supervised learning (Peng et al., 2020; Radosavovic et al.,
 121 2021; Zhao et al., 2023a; Tie et al., 2025; Chi et al., 2023). However, many recent IL methods rely on
 122 large-scale datasets to learn robust manipulation policies (Radosavovic et al., 2021; Peng et al., 2020).
 123 To improve data efficiency, several works (Tie et al., 2025; Yang et al., 2024a; Wang et al., 2024a)
 124 incorporate equivariance into policy architectures, thereby enhancing spatial generalization. Other
 125 approaches have explored multi-modal fusion of 3D vision, language instructions, and proprioception
 126 (Gervet et al., 2023; Shridhar et al., 2023; Zhang et al., 2023; 2024; Ke et al., 2024a). Despite
 127 their success, these methods typically predict discrete keyframes rather than continuous control
 128 trajectories (e.g., PerAct (Shridhar et al., 2023), Act3D (Gervet et al., 2023), Chained Diffuser (Xian
 129 et al., 2023), and 3D Diffuser Actor (Ke et al., 2024a)), limiting their effectiveness in long-horizon
 130 or fine-grained manipulation tasks. A method most closely related to ours is GenDP (Wang et al.,
 131 2024c), which computes dense semantic fields by measuring cosine similarity between 2D image
 132 features and scene observations. This enables category-level generalization, but suffers from two key
 133 limitations: (1) its reliance on 2D features from DINOv2 introduces inconsistencies across views;
 134 and (2) its semantic representations lack the granularity needed to identify functionally relevant
 135 object parts, which are critical for manipulation. In contrast, our method introduces a 3D-native
 136 fine-grained feature field that is function-aware, allowing for more accurate localization of interactive
 137 parts. Based on this representation, we develop a manipulation policy that not only requires fewer
 138 demonstrations but also generalizes across unseen object categories—addressing both data efficiency
 139 and generalization in robotic manipulation.

3 METHOD

140 In this section, we cover the different components of our approach, as shown in 2. Initially, we present
 141 an overview of Part-Aware 3D Feature Field (PA3FF), and data strategy, training process, along with
 142 model architecture in section 3.1. Subsequently, we explore how to leverage this 3D feature field to
 143 achieve generalizable articulated object manipulation learning in section 3.2.

3.1 PART-AWARE 3D FEATURE FIELD

144 **Problem Formulation.** Our objective is to design a 3D and part-awareness feature for generalizable
 145 articulated object manipulation. Given an input point cloud $\mathcal{P} = \{\mathbf{p}_i \in \mathbb{R}^3\}_{i=1}^N$, this model predicts
 146 a continuous 3D feature field $f : \mathbb{R}^3 \rightarrow \mathbb{R}^n$ that encodes the part structure and their hierarchy in a
 147 feedforward manner. This feature field assigns each point $\mathbf{p} \in \mathcal{P}$ an n -dimensional latent feature
 148 vector $f(\mathbf{p})$, resulting in a per-point embedding of the input point clouds. The notion of parts is
 149 captured by the proximity of features in this latent space: points \mathbf{p}_a and \mathbf{p}_b that belong to the same
 150 part should have similar features, i.e., $f(\mathbf{p}_a) \approx f(\mathbf{p}_b)$.

151 **Backbone for 3D Feature Extraction.** In this stage, we aim to get a 3D feature extraction backbone
 152 that leverages the geometric cues of 3D objects and learns 3D priors from a large-scale 3D dataset.
 153 Unlike prior work He et al. (2024); Yang et al. (2024b); Kim et al. (2024a); Liu et al. (2023); Zhou
 154 et al. (2023) that relies on per-shape optimization to lift or distill 2D predictions or priors, we
 155 instead leverage Sonata (Wu et al., 2025), a self-supervised pre-trained Point Transformer V3 (Wu
 156 et al., 2024). We then employ Sonata with its pre-trained weights as our feature extractor $f(\mathbf{p})$ to
 157 extract multi-scale features from point clouds. The key advantage of our approach lies in addressing
 158 limitations present in prior methods that rely on 2D feature distillation. First, multi-view fusion in
 159 previous approaches is computationally slow and suffers from the problem of inconsistent features

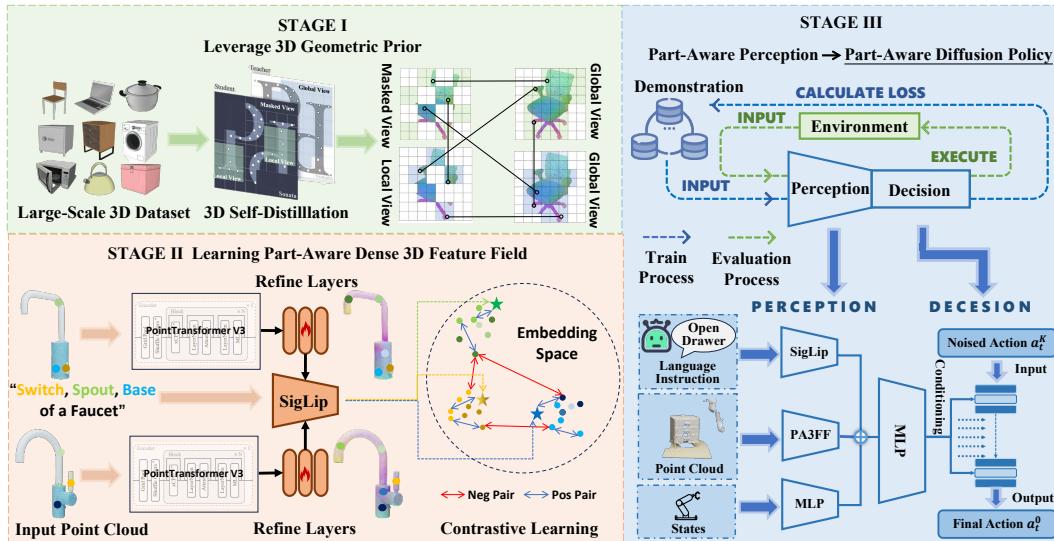


Figure 2: Overview of our Learning Framework. (1) *Pretraining the PTv3 backbone to extract part-aware 3D features.* (2) *Feature refinement via contrastive learning across objects to enhance part-level consistency and distinctiveness.* (3) *Downstream usage by integrating the refined features into a diffusion policy for action generation.*

across views. Second, these methods typically sacrifice spatial resolution for semantic quality in 2D features—for example, ViT-family models process image tokens as patches, resulting in much lower-resolution feature maps (e.g., 14x smaller in DINOv2), leading to a significant loss of spatial information. In contrast, Sonata allows for 3D dense feature extraction that maintains both geometric accuracy and semantic information. This approach offers several advantages: (a) efficient, feedforward inference; (b) consistent and complete 3D feature fields that generalize well across objects; and (c) per-point dense features that capture accurate geometric cues. However, Sonata was originally trained on scene-level data and is not directly tailored for object- or part-level representation learning. Specifically, its backbone, PointTransformer v3 (PTv3), is designed for large-scale scenes, using aggressive downsampling to expand receptive fields and reduce computational cost. In contrast, object-level inputs are smaller in both point count and spatial extent, making such downsampling suboptimal. To adapt Sonata for our task, we remove most downsampling layers in PTv3 and instead deepen the network by stacking additional transformer blocks, which enhances detail preservation and improves feature abstraction. Importantly, our overall framework is model-agnostic and can accommodate more advanced 3D feature extraction. More details can be found in Appendix A

Learning Part-Aware Dense 3D Feature Field. Building on the promising 3D priors we have obtained, we aim to enhance these representations by incorporating part-aware semantic features. To achieve this, we introduce a contrastive learning framework that effectively establishes consistent relationships between 3D part-level features and their corresponding semantic counterparts.

We design two complementary loss functions to achieve this goal. The first one is **geometric loss**, which focuses on the spatial relationships between points within the same part and between different parts. It encourages the model to bring points from the same part closer together in the feature space while pushing points from different parts apart. Given a set of N feature/label pairs $\{f_k, a_k\}_{k=1\dots N}$, and assumes there are N_{a_k} samples sharing label a_k . The geometric loss is derived from the Supervised Contrastive Loss (Khosla et al., 2021), a widely-used method in contrastive learning. The geometric loss is defined as:

$$\mathcal{L}_{Geo} = \sum_{i=1}^N \frac{-1}{N_{a_i} - 1} \sum_{j=1}^N \mathbf{1}_{i \neq j} \cdot \mathbf{1}_{a_i = a_j} \cdot \log \frac{\exp(\mathbf{f}_i \cdot \mathbf{f}_j / \tau)}{\sum_{k=1}^N \mathbf{1}_{i \neq k} \cdot \exp(\mathbf{f}_i \cdot \mathbf{f}_k / \tau)} \quad (1)$$

where $\tau \in \mathbb{R}^+$ denotes the balancing coefficient in SupCon.

In addition to geometric alignment, we introduce a **semantic loss**, which aligns point-level features with semantic representations derived from part names. Specifically, we leverage SigLip (Zhai et al., 2023) to encode the part names from the dataset into semantic vectors. These semantic vectors are then used as targets for aligning point features through InfoNCE Loss (van den Oord et al., 2019), which encourages the model to map point-level features to their corresponding semantic representations.

Let m represent the number of distinct part names in the current object category, denoted as $\{s_1, s_2, \dots, s_m\}$. These part names are encoded using the SigLIP text encoder to obtain semantic representations $\mathbf{x}_k = \text{SigLip}(s_k)$, $k = 1 \dots m$. Given a set of N feature/label pairs $\{\mathbf{f}_k, a_k\}_{k=1 \dots N}$, where $a_k \in 1, \dots, m$ is the index of the ground-truth part name for the k -th point feature, just like above. The semantic Loss is defined as:

$$\mathcal{L}_{Sem} = \sum_{i=1}^N -\log \frac{\exp(\mathbf{f}_i \cdot \mathbf{x}_{a_i}/\tau)}{\sum_{k=1}^m \exp(\mathbf{f}_i \cdot \mathbf{x}_k/\tau)} \quad (2)$$

where $\tau \in \mathbb{R}^+$ denotes the balancing coefficient in InfoNCE.

The total loss used for training combines both the geometric loss and the semantic loss:

$$\mathcal{L}_{total} = \mathcal{L}_{Geo} + \mathcal{L}_{Sem} \quad (3)$$

This combined loss function ensures that the model learns features that are both geometrically consistent within parts and semantically aligned with their corresponding part names. To further refine the feature representations, we propose a lightweight feature refinement network, consisting of a shallow per-point MLP. This network processes the output from the Sonata model and uses the total loss function to guide the learning process, as shown in Fig 2.

3.2 PART-AWARE DIFFUSION POLICY

Problem Formulation. Part-Aware Diffusion Policy is a diffusion-based model for action generation (Chi et al., 2023; Tie et al., 2025) that takes 3D point cloud observations and robot proprioceptive state as input to predict future action sequences (action chunks). We formulate visuomotor control as modeling the conditional distribution $p(A_t | \mathbf{o}_t)$, where the observation at time t is $\mathbf{o}_t = [\mathbf{P}_t^1, \dots, \mathbf{P}_t^n, \mathbf{q}_t]$, with \mathbf{P}_t^i representing the point cloud from the i th camera view and \mathbf{q}_t the proprioceptive state. The predicted action chunk is $A_t = [a_t, a_{t+1}, \dots, a_{t+H-1}]$. We train a Denoising Diffusion Probabilistic Model (DDPM) (Ho et al., 2020) and use Denoising Diffusion Implicit Model (DDIM) (Song et al., 2022) for accelerated inference sampling. The denoising process is defined as:

$$\mathbf{a}_t^{k-1} = \frac{\sqrt{\bar{\beta}^{k-1}} \gamma^k}{1 - \bar{\beta}^k} \mathbf{a}_t^0 + \frac{\sqrt{\beta^k} (1 - \bar{\beta}^{k-1})}{1 - \bar{\beta}^k} \mathbf{a}_t^k + \tau^k \mathbf{v}, \quad (4)$$

where $\{\beta^k\}_{k=1}^K$ and $\{\tau^k\}_{k=1}^K$ are scalar coefficients from a predefined noise schedule, $\gamma^k := 1 - \beta^k$, and $\bar{\beta}^{k-1} := \prod_{i=1}^{k-1} \beta^i$. The noise term is $\mathbf{v} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ when $k > 1$; otherwise, $\bar{\beta}^{k-1} = 1$ and $\mathbf{v} = \mathbf{0}$. The model is trained by minimizing the mean squared error (MSE) between the ground-truth action \mathbf{a}_t and the model's prediction:

$$\mathcal{L}(\phi) := \text{MSE}(\mathbf{a}_t, D_\theta(\mathbf{o}_t, \tilde{\mathbf{a}}_t, k)), \quad (5)$$

where

$$\tilde{\mathbf{a}}_t := \sqrt{\bar{\beta}^k} \mathbf{a}_t + \sqrt{1 - \bar{\beta}^k} \epsilon, \quad \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad k \sim \text{Uniform}(\{1, \dots, K\}). \quad (6)$$

Policy Design. We employ our Part-Aware 3D Feature Field (PA3FF) as a frozen backbone to extract point cloud embeddings. These embeddings are then fed into a trainable Transformer encoder to aggregate per-point features into a global representation. Since the features provided by our backbone are semantically meaningful, we use the semantic embedding of the task-critical part name as the CLS token to guide this aggregation. Next, we concatenate the resulting global scene feature with the agent's pos and pass it through a two-layer MLP to reduce its length, producing the encoder's final output. Conditioned on the compact representation, a diffusion action head outputs the robot action.



Figure 3: Task illustrations. We evaluate our model on eight downstream tasks.

4 EXPERIMENT

We systematically evaluate PA3FF and PADP through both simulation and real-world experiments, aiming to address the following questions: (1) How does the performance of our method compare to previous imitation approaches? (2) How well does our method generalize under object and environmental perturbations? (3) What factors contribute to the generalization of our method to novel instances? (4) Beyond imitation learning, what additional applications can PA3FF facilitate?

4.1 EXPERIMENTAL SETUP

We benchmark PADP in both simulated and real-world environments. The simulated environments serve as a controlled platform to ensure reproducible and fair comparisons. The real-world experiments demonstrate the method’s applicability to real-world settings.

Setup. In simulation, we conduct multi-task training on the PartInstruct benchmark (Yin et al., 2025), which focuses on part-level fine-grained manipulation tasks. For real-world experiments, we use a Franka Emika Panda robotic arm equipped with UMI fingers (Chi et al., 2024b) replacing the standard parallel gripper. Perception is handled by three Intel RealSense D415 depth cameras positioned around the workspace. Figure 9 illustrates our real-world setup and experimental objects.

Tasks and Metrics. We design 8 real-world tasks spanning diverse manipulation scenarios (Figure 3). Each method is evaluated over 10 trials per task with randomized object layouts. Simulation follows PartInstruct’s five-level protocol testing generalization across object states (OS), object instances (OI), part combinations in the same task type (TP), task categories (TC), and object categories (OC). More details can be found in Appendix C.2.

Data Collections. For the real-world experiments, we collect demonstrations by human teleoperation. The Franka arm and the gripper are teleoperated by the keyboard. Since our tasks contain more than one stage and include two robots and various objects, making the process of demonstration collection very time-consuming, we only provide 30 demonstrations for each task. For all six tasks, object poses are randomly initialized on the table. The action space contains the end-effector pose and gripper state, while observations include RGB images and corresponding depth images captured by three Intel RealSense D415 depth cameras. For simulation tasks, we leverage the demonstrations provided from Part-Instruct (Yin et al., 2025). More detailed settings can be found in Appendix E.

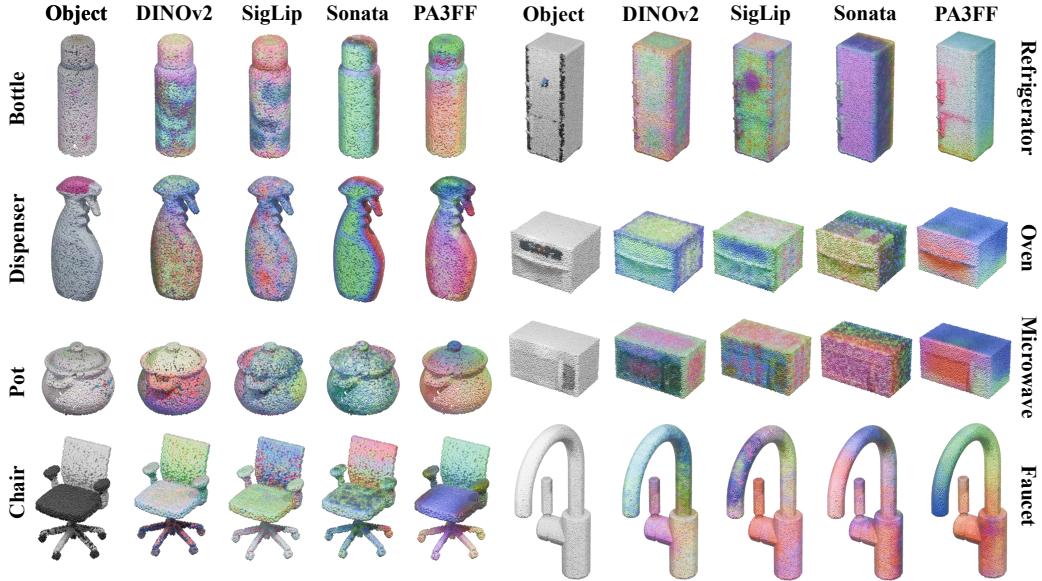
Baselines. In order to comprehensively evaluate PADP, we carefully select six baselines. These include image-to-action imitation learning baselines, *Diffusion Policy (DP)* (Chi et al., 2023). We also compare with models that have been specifically designed for 3D object manipulation including *Act3D* (Gervet et al., 2023), *RVT2* (Goyal et al., 2024), *3D Diffuser Actor (3D-DA)* (Ke et al., 2024b), *GenDP* (Wang et al., 2024c), *DP3* (Ze et al., 2024). These baselines show different 2D and 3D representations for policy. For DP, DP3, and GenDP, we add a language-conditioning module in the same manner as PADP to fuse language instructions. Details can be found in Appendix C.3

324 **4.2 COMPARISON WITH BASELINES**
 325

326 Table 1: Simulated results across five test sets. The best-performing results are highlighted in bold
 327

Method	Test 1 (OS)	Test 2 (OI)	Test 3 (TP)	Test 4 (TC)	Test 5 (OC)	Average
Act 3D (Gervet et al., 2023)	6.25±1.8	5.68±1.7	4.55±1.6	0.0	2.08±2.1	3.88±1.8
RVT2 (Goyal et al., 2024)	4.55±2.0	4.55±2.0	6.36±2.3	0.91±0.9	3.33±3.3	4.04±2.1
3D-DA (Ke et al., 2024a)	8.08±2.7	5.05±2.2	4.04±1.9	0.0	3.70±3.6	4.26±1.0
DP (Chi et al., 2023)	7.27±1.8	8.64±1.9	8.18±1.8	3.75±2.1	6.67±3.2	5.96±2.2
DP3 (Ze et al., 2024)	23.18±2.8	23.18±2.8	18.18±2.6	7.73±1.8	6.67±3.2	15.40±2.6
GenDP (Wang et al., 2024c)	24.34±2.1	23.36±2.3	24.53±1.9	10.00±2.0	14.61±2.1	19.36±2.7
Ours	36.76±2.3	34.33±3.6	32.45±1.6	13.75±2.0	26.67±3.2	28.79±2.5

336 **Comparision PA3FF with other foundation features.** For an intuitive understanding of PA3FF,
 337 Figure 4 illustrates the feature fields of various objects. DINOv2 and SigLip, two 2D methods, use
 338 rendered images from 16 views of the mesh as input. After each image is encoded, the resulting
 339 features are remapped onto the mesh surface to generate a feature map. Sonata and our PA3FF,
 340 however, directly utilize point clouds sampled from the mesh as input to generate feature maps. The
 341 feature maps shown here for the 2D methods represent the result of mapping mesh features onto
 342 the point clouds input to Sonata and PA3FF. Compared to 2D foundation features like DINOv2
 343 and SigLIP, our approach leverages the continuity of Sonata features, avoiding common **multi-view**
 344 **consistency** issues when aggregating 2D features from different views. As a result, the generated
 345 feature fields are smoother and less noisy (as shown in the faucet example). Our method also
 346 highlights key functional parts, such as microwave and refrigerator handles. Another limitation of 2D
 347 methods is their difficulty in **capturing small or thin parts**, which may occupy less than a patch
 348 or pixel in the 2D images and fail to be adequately represented (e.g., the refrigerator handle on the
 349 right). In contrast to Sonata, our approach explicitly promotes intra-part feature consistency and inter-
 350 part distinctiveness within specific object categories, leading to more semantically meaningful and
 351 discriminative part-level representations. More feature visualization can be found in Appendix A.2



371 Figure 4: The feature field visualizations of PA3FF and other foundation features.
 372

373 **Performance and Comparison.** Table 2 presents the main results on real-world tasks. PADP
 374 significantly outperforms several strong baselines across all tasks, by achieving a mean success rate
 375 of 58.75% under unseen objects, compared to the highest success rate of 35% achieved by baselines.
 376 Figure 9 presents snapshots of the real-world experiments. Table 1 shows the results on simulation
 377 tasks. Consistent with the real-world results, the simulation results also demonstrate that PADP
 enhances policy generalization.

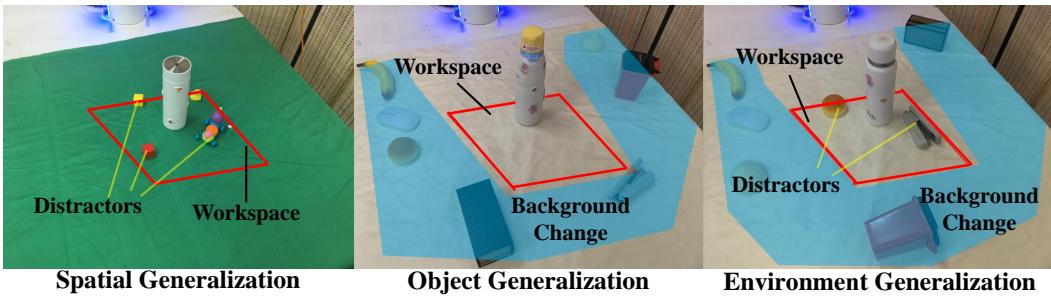
Table 2: Real-world task success rates across different methods (train/test). Each task is evaluated with 10 trials under randomized initial conditions. The best-performing results are highlighted in bold

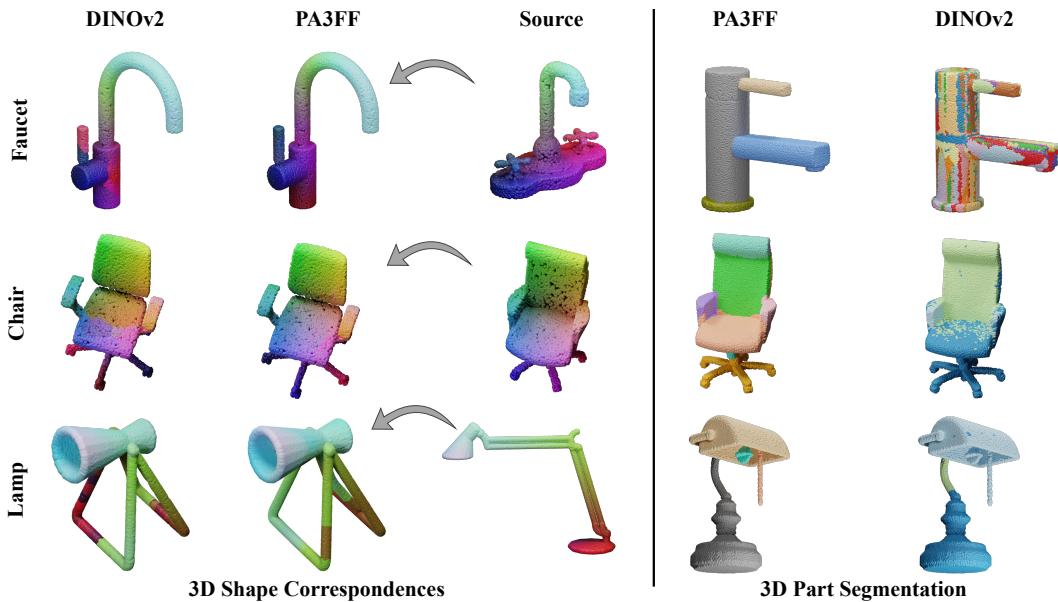
Method/Task	Pulling lid of pot		Open drawer		Close box		Close lid of laptop	
	Train	Test	Train	Test	Train	Test	Train	Test
DP (Chi et al., 2023)	4/10	2/10	1/10	1/10	2/10	1/10	4/10	3/10
DP3 (Ze et al., 2024)	6/10	4/10	4/10	3/10	3/10	3/10	5/10	5/10
GenDP (Wang et al., 2024c)	7/10	6/10	5/10	5/10	3/10	3/10	6/10	4/10
PADP (Ours)	8/10	6/10	6/10	6/10	5/10	5/10	7/10	7/10
Method/Task	Open microwave		Open bottle		Put lid on kettle		Press dispenser	
	Train	Test	Train	Test	Train	Test	Train	Test
DP (Chi et al., 2023)	1/10	0/10	5/10	2/10	1/10	0/10	0/10	0/10
DP3 (Ze et al., 2024)	3/10	1/10	4/10	3/10	3/10	1/10	1/10	1/10
GenDP (Wang et al., 2024c)	4/10	3/10	5/10	4/10	4/10	2/10	2/10	1/10
PADP (Ours)	6/10	5/10	8/10	6/10	6/10	5/10	4/10	3/10

Table 3: Generalization evaluation of the *Open Bottle* task (10 trials).

Method	Completion Rate (%)			
	Original	Disturbance		
		Spatial	Object	Environment
DP	50	40 <small>↓10</small>	10 <small>↓40</small>	0 <small>↓50</small>
DP3	40	20 <small>↓20</small>	20 <small>↓20</small>	10 <small>↓30</small>
GenDP	50	20 <small>↓30</small>	30 <small>↓20</small>	30 <small>↓20</small>
PADP (ours)	80	70 <small>↓10</small>	60 <small>↓20</small>	60 <small>↓20</small>

Following (Wang et al., 2025), we evaluate generalization across three dimensions: *Spatial*, *Object* and *Environment*. **(1) Spatial Generalization:** When object poses change, baseline methods struggle to locate handles in the Open Microwave task. PADP consistently identifies correct grasping positions (Figure 15 in Appendix D) because its part-aware feature field effectively locates functional parts and understands object geometry. **(2) Object Generalization:** As demonstrated in Table 2, PADP maintains robust performance on unseen objects, while DP and DP3 fail on new instances. GenDP leverages semantic fields for improved generalization, but PADP surpasses all methods through PA3FF’s precise feature representation. For example, in the Open Microwave task (see Figure 15 in Appendix D), Despite varying appearances across microwave models, PA3FF identifies shared functional structures (handles, bodies), enabling consistent manipulation regardless of shape or pose variations. **(3) Environment Generalization:** We evaluate robustness across four scenarios (Figure 5): original environment (*Situation 0*), added distractors (*Situation 1*), changed background (*Situation 2*), and combined changes (*Situation 3*). For the Open Bottle task, PADP maintains high performance across all conditions, while baselines degrade significantly in complex scenarios (Table 3). DP fails with background changes due to image dependency, DP3 resists color changes but struggles with distractors, and GenDP, despite outperforming other baselines, cannot handle the combined challenges of *Situation 3*.

Figure 5: Generalization test set of the *Open Bottle* task.

432 4.3 VARIOUS DOWNSTREAM APPLICATIONS.
433434 We evaluated the properties of the learned feature field in various applications, including part
435 decomposition, 3D shape correspondences, and feature field consistency.
436455 Figure 6: PA3FF exhibits consistency across shapes, enabling applications such as correspondence
456 learning and part segmentation.
457458
459 **3D Shape Correspondences.** The cross-shape consistency embedded in PA3FF provides a robust
460 prior for fine-grained point-to-point correspondence learning. As a promising application, we use
461 Functional Maps Ovsjanikov et al. (2012) to establish correspondences between a source and a target
462 shape. To begin, we initialize the correspondences by finding nearest neighbors in the PA3FF feature
463 space. These initial correspondences are then refined using Smooth Discrete Optimization Magnet
464 et al. (2022), which iteratively solves functional maps in a coarse-to-fine fashion to recover a smooth
465 and accurate point-to-point mapping. As shown in Figure 6, we compare the performance of this
466 method against DINOv2 Oquab et al. (2023) features. In particular, PA3FF excels in providing
467 precise correspondences, even in challenging cases where the shapes differ significantly in topology
468 or pose. Beyond capturing shape similarity, PA3FF is capable of encoding functional semantics of
469 parts—allowing it to match parts based on their function rather than just appearance—while still
470 maintaining smoothness in the resulting correspondences.471 **3D Part Segmentation.** PA3FF learns a part hierarchy implicitly through contrastive learning on
472 diverse 3D data. This hierarchical structure can also be explicitly derived using agglomerative
473 clustering. Figure 6 shows the results obtained from the clustering of the DINOv2 and PA3FF feature
474 maps using identical parameters. Table 4 provide quantitative results of segmentation performance on
475 the PartNet-Ensembled (PartNetE) (Liu et al., 2023) dataset. PA3FF successfully identifies significant
476 relationships between parts, a feature that can be leveraged in a range of practical applications.
477478 5 CONCLUSION
479480 We propose Part-Aware 3D Feature Field (PA3FF), a novel 3D feature representation that enhances
481 generalization for articulated object manipulation by focusing on functional parts. Combined with
482 the Part-Aware Diffusion Policy (PADP), an imitation learning framework, PA3FF improves sample
483 efficiency and generalization. Experimental results show that PADP outperforms existing 2D and 3D
484 representations, including CLIP, DINOv2, and Grounded-SAM, in both simulated and real-world
485 tasks. Beyond imitation learning, PA3FF enables a variety of downstream tasks, demonstrating its
versatility and effectiveness in robotic manipulation.

486
487 ETHICS STATEMENT488
489 This work does not involve human subjects, personally identifiable information, or sensitive data.
490 The datasets used are publicly available, and all experiments comply with the ICLR Code of Ethics.
491492 REPRODUCIBILITY STATEMENT
493494 We have made extensive efforts to ensure reproducibility. The detailed methodology of our proposed
495 approach is presented in Section 3, while the experimental settings, including training procedures and
496 evaluation protocols, are described in Section 4. To further support reproducibility, we plan to release
497 the complete source code and instructions upon the acceptance of this paper.
498499 REFERENCES
500

- 501 Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai,
502 Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey
503 Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James
504 Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0 : A vision-language-
505 action flow model for general robot control, 2024. URL <https://arxiv.org/abs/2410.24164>.
- 506
507 Xiaotong Chen, Kaizhi Zheng, Zhen Zeng, Cameron Kisailus, Shreshtha Basu, James Cooney, Jana
508 Pavlasek, and Odest Chadwicke Jenkins. Manipulation-oriented object perception in clutter through
509 affordance coordinate frames. In *2022 IEEE-RAS 21st International Conference on Humanoid
510 Robots (Humanoids)*, pp. 186–193. IEEE, 2022.
- 511 Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake,
512 and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The
513 International Journal of Robotics Research*, pp. 02783649241273668, 2023.
- 514 Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake,
515 and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion, 2024a. URL
516 <https://arxiv.org/abs/2303.04137>.
- 517 Cheng Chi, Zhenjia Xu, Chuer Pan, Eric Cousineau, Benjamin Burchfiel, Siyuan Feng, Russ Tedrake,
518 and Shuran Song. Universal manipulation interface: In-the-wild robot teaching without in-the-wild
519 robots, 2024b. URL <https://arxiv.org/abs/2402.10329>.
- 520 Norman Di Palo and Edward Johns. On the effectiveness of retrieval, alignment, and replay in
521 manipulation. *IEEE Robotics and Automation Letters*, 2024.
- 522 Stephanie Fu, Mark Hamilton, Laura Brandt, Axel Feldman, Zhoutong Zhang, and William T.
523 Freeman. Featup: A model-agnostic framework for features at any resolution, 2024. URL
524 <https://arxiv.org/abs/2403.10516>.
- 525 Theophile Gervet, Zhou Xian, Nikolaos Gkanatsios, and Katerina Fragkiadaki. Act3d: 3d feature field
526 transformers for multi-task robotic manipulation. In *7th Annual Conference on Robot Learning*,
527 2023.
- 528 Ankit Goyal, Valts Blukis, Jie Xu, Yijie Guo, Yu-Wei Chao, and Dieter Fox. Rvt-2: Learning precise
529 manipulation from few demonstrations. *arXiv preprint arXiv:2406.08545*, 2024.
- 530 Haodi He, Colton Stearns, Adam W. Harley, and Leonidas J. Guibas. View-consistent hierarchical 3d
531 segmentation using ultrametric feature fields, 2024. URL <https://arxiv.org/abs/2405.19678>.
- 532 Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image
533 recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*,
534 pp. 770–778, 2016.

- 540 Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. URL
 541 <https://arxiv.org/abs/2006.11239>.
- 542
- 543 Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess,
 544 Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh,
 545 Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin
 546 LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z.
 547 Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner,
 548 Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a
 549 vision-language-action model with open-world generalization, 2025. URL <https://arxiv.org/abs/2504.16054>.
- 550
- 551 Li Jiang, Hengshuang Zhao, Shaoshuai Shi, Shu Liu, Chi-Wing Fu, and Jiaya Jia. Pointgroup:
 552 Dual-set point grouping for 3d instance segmentation, 2020. URL <https://arxiv.org/abs/2004.01658>.
- 553
- 554 Tsung-Wei Ke, Nikolaos Gkanatsios, and Katerina Fragkiadaki. 3d diffuser actor: Policy diffusion
 555 with 3d scene representations. *Arxiv*, 2024a.
- 556
- 557 Tsung-Wei Ke, Nikolaos Gkanatsios, and Katerina Fragkiadaki. 3d diffuser actor: Policy diffusion
 558 with 3d scene representations, 2024b. URL <https://arxiv.org/abs/2402.10885>.
- 559
- 560 Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. Lerf: Language
 561 embedded radiance fields. *arXiv preprint arXiv:2303.09553*, 2023.
- 562
- 563 Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron
 564 Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning, 2021. URL <https://arxiv.org/abs/2004.11362>.
- 565
- 566 Chung Min Kim, Mingxuan Wu, Justin Kerr, Ken Goldberg, Matthew Tancik, and Angjoo Kanazawa.
 567 Garfield: Group anything with radiance fields, 2024a. URL <https://arxiv.org/abs/2401.09419>.
- 568
- 569 Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair,
 570 Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source
 571 vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024b.
- 572
- 573 Mia Kokic, Johannes A Stork, Joshua A Haustein, and Danica Kragic. Affordance detection for
 574 task-specific grasping using deep learning. In *2017 IEEE-RAS 17th International Conference on
 Humanoid Robotics (Humanoids)*, pp. 91–98. IEEE, 2017.
- 575
- 576 Xingyu Lin, John So, Sashwat Mahalingam, Fangchen Liu, and Pieter Abbeel. Spawnnnet: Learning
 577 generalizable visuomotor skills from pre-trained networks. *arXiv preprint arXiv:2307.03567*,
 2023.
- 578
- 579 Minghua Liu, Yinhao Zhu, Hong Cai, Shizhong Han, Zhan Ling, Fatih Porikli, and Hao Su. Partslip:
 580 Low-shot part segmentation for 3d point clouds via pretrained image-language models, 2023. URL
<https://arxiv.org/abs/2212.01558>.
- 581
- 582 Robin Magnet, Jing Ren, Olga Sorkine-Hornung, and Maks Ovsjanikov. Smooth non-rigid shape
 583 matching via effective dirichlet energy optimization. In *2022 International Conference on 3D
 Vision (3DV)*, pp. 495–504. IEEE, 2022.
- 584
- 585 Kaichun Mo, Shilin Zhu, Angel X. Chang, Li Yi, Subarna Tripathi, Leonidas J. Guibas, and Hao
 586 Su. Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object
 587 understanding, 2018. URL <https://arxiv.org/abs/1812.02713>.
- 588
- 589 Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov,
 590 Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaeldin El-Nouby, et al. Dinov2: Learning
 591 robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.
- 592
- 593 Maks Ovsjanikov, Mirela Ben-Chen, Justin Solomon, Adrian Butscher, and Leonidas Guibas. Func-
 594 tional maps: a flexible representation of maps between shapes. *ACM Transactions on Graphics
 (ToG)*, 31(4):1–11, 2012.

- 594 David Paulius, Yongqiang Huang, Roger Milton, William D Buchanan, Jeanine Sam, and Yu Sun.
 595 Functional object-oriented network for manipulation learning. In *2016 IEEE/RSJ International*
 596 *Conference on Intelligent Robots and Systems (IROS)*, pp. 2655–2662. IEEE, 2016.
- 597
- 598 Xue Bin Peng, Erwin Coumans, Tingnan Zhang, Tsang-Wei Lee, Jie Tan, and Sergey Levine.
 599 Learning agile robotic locomotion skills by imitating animals. *arXiv preprint arXiv:2004.00784*,
 600 2020.
- 601 Charles R. Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning
 602 on point sets in a metric space, 2017. URL <https://arxiv.org/abs/1706.02413>.
- 603
- 604 Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal,
 605 Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual
 606 models from natural language supervision. In *International conference on machine learning*, pp.
 607 8748–8763. PMLR, 2021.
- 608
- 609 Ilija Radosavovic, Xiaolong Wang, Lerrel Pinto, and Jitendra Malik. State-only imitation learning for
 610 dexterous manipulation. In *2021 IEEE/RSJ International Conference on Intelligent Robots and*
611 Systems (IROS), pp. 7865–7871. IEEE, 2021.
- 612
- 613 Adam Rashid, Satvik Sharma, Chung Min Kim, Justin Kerr, Lawrence Yunliang Chen, Angjoo
 614 Kanazawa, and Ken Goldberg. Language embedded radiance fields for zero-shot task-oriented
 615 grasping. In *7th Annual Conference on Robot Learning*, 2023.
- 616
- 617 Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang,
 618 Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing
 619 Jiang, and Lei Zhang. Grounded sam: Assembling open-world models for diverse visual tasks,
 620 2024. URL <https://arxiv.org/abs/2401.14159>.
- 621
- 622 William Shen, Ge Yang, Alan Yu, Jensen Wong, Leslie Pack Kaelbling, and Phillip Isola. Distilled
 623 feature fields enable few-shot manipulation. 2023.
- 624
- 625 Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Perceiver-actor: A multi-task transformer for
 626 robotic manipulation. In *Conference on Robot Learning*, pp. 785–799. PMLR, 2023.
- 627
- 628 Habib Slim, Xiang Li, Yuchen Li, Mahmoud Ahmed, Mohamed Ayman, Ujjwal Upadhyay,
 629 Ahmed Abdelreheem, Arpit Prajapati, Suhail Pothigara, Peter Wonka, and Mohamed Elhoseiny.
 630 3dcompat⁺⁺: An improved large-scale 3d vision dataset for compositional recognition, 2025.
 631 URL <https://arxiv.org/abs/2310.18511>.
- 632
- 633 Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022. URL
 634 <https://arxiv.org/abs/2010.02502>.
- 635
- 636 Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep
 637 Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, et al. Octo: An open-source generalist robot
 638 policy. *arXiv preprint arXiv:2405.12213*, 2024.
- 639
- 640 Chenrui Tie, Yue Chen, Ruihai Wu, Boxuan Dong, Zeyi Li, Chongkai Gao, and Hao Dong. Et-seed:
 641 Efficient trajectory-level se(3) equivariant diffusion policy, 2025. URL <https://arxiv.org/abs/2411.03990>.
- 642
- 643 Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive
 644 coding, 2019. URL <https://arxiv.org/abs/1807.03748>.
- 645
- 646 Thang Vu, Kookhoi Kim, Tung M. Luu, Xuan Thanh Nguyen, and Chang D. Yoo. Softgroup for
 647 3d instance segmentation on point clouds, 2022. URL <https://arxiv.org/abs/2203.01509>.
- 648
- 649 Dian Wang, Stephen Hart, David Surovik, Tarik Kelestemur, Haojie Huang, Haibo Zhao, Mark
 650 Yeatman, Jiuguang Wang, Robin Walters, and Robert Platt. Equivariant diffusion policy, 2024a.
 651 URL <https://arxiv.org/abs/2407.01812>.

- 648 Qianxu Wang, Haotong Zhang, Congyue Deng, Yang You, Hao Dong, Yixin Zhu, and Leonidas
 649 Guibas. Sparsedff: Sparse-view feature distillation for one-shot dexterous manipulation, 2024b.
 650 URL <https://arxiv.org/abs/2310.16838>.
- 651 Shengjie Wang, Jiacheng You, Yihang Hu, Jiongye Li, and Yang Gao. Skil: Semantic keypoint
 652 imitation learning for generalizable data-efficient manipulation, 2025. URL <https://arxiv.org/abs/2501.14400>.
- 653 Yian Wang, Ruihai Wu, Kaichun Mo, Jiaqi Ke, Qingnan Fan, Leonidas J Guibas, and Hao Dong.
 654 Adaafford: Learning to adapt manipulation affordance for 3d articulated objects via few-shot
 655 interactions. In *European conference on computer vision*, pp. 90–107. Springer, 2022.
- 656 Yixuan Wang, Guang Yin, Binghao Huang, Tarik Kelestemur, Jiuguang Wang, and Yunzhu Li.
 657 Gendp: 3d semantic fields for category-level generalizable diffusion policy, 2024c. URL <https://arxiv.org/abs/2410.17488>.
- 658 Bowen Wen, Wenzhao Lian, Kostas Bekris, and Stefan Schaal. You only demonstrate once: Category-
 659 level manipulation from single visual demonstration. *Robotics: Science and Systems* 2022, 2022.
- 660 Ruihai Wu and Yan Zhao. Vat-mart: Learning visual action trajectory proposals for manipulating 3d
 661 articulated objects. In *International Conference on Learning Representations (ICLR)*, 2022, 2022.
- 662 Ruihai Wu, Chuanruo Ning, and Hao Dong. Learning foresightful dense visual affordance for
 663 deformable object manipulation. In *IEEE International Conference on Computer Vision (ICCV)*,
 664 2023.
- 665 Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong
 666 He, and Hengshuang Zhao. Point transformer v3: Simpler, faster, stronger, 2024. URL <https://arxiv.org/abs/2312.10035>.
- 667 Xiaoyang Wu, Daniel DeTone, Duncan Frost, Tianwei Shen, Chris Xie, Nan Yang, Jakob Engel,
 668 Richard Newcombe, Hengshuang Zhao, and Julian Straub. Sonata: Self-supervised learning of
 669 reliable point representations, 2025. URL <https://arxiv.org/abs/2503.16429>.
- 670 Zhou Xian, Nikolaos Gkanatsios, Theophile Gervet, Tsung-Wei Ke, and Katerina Fragkiadaki.
 671 Chaineddiffuser: Unifying trajectory diffusion and keypose prediction for robotic manipulation. In
 672 *7th Annual Conference on Robot Learning*, 2023.
- 673 Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao
 674 Jiang, Yifu Yuan, He Wang, Li Yi, Angel X. Chang, Leonidas J. Guibas, and Hao Su. Sapien: A
 675 simulated part-based interactive environment, 2020. URL <https://arxiv.org/abs/2003.08515>.
- 676 Jingyun Yang, Zi ang Cao, Congyue Deng, Rika Antonova, Shuran Song, and Jeannette Bohg.
 677 Equibot: Sim(3)-equivariant diffusion policy for generalizable and data efficient learning, 2024a.
 678 URL <https://arxiv.org/abs/2407.01479>.
- 679 Yunhan Yang, Yukun Huang, Yuan-Chen Guo, Liangjun Lu, Xiaoyang Wu, Edmund Y. Lam, Yan-
 680 Pei Cao, and Xihui Liu. Sampart3d: Segment any part in 3d objects, 2024b. URL <https://arxiv.org/abs/2411.07184>.
- 681 Yifan Yin, Zhengtao Han, Shivam Aarya, Jianxin Wang, Shuhang Xu, Jiawei Peng, Angtian Wang,
 682 Alan Yuille, and Tianmin Shu. Partinstruct: Part-level instruction following for fine-grained robot
 683 manipulation, 2025. URL <https://arxiv.org/abs/2505.21652>.
- 684 Yanjie Ze, Ge Yan, Yueh-Hua Wu, Annabella Macaluso, Yuying Ge, Jianglong Ye, Nicklas Hansen,
 685 Li Erran Li, and Xiaolong Wang. Gnfactor: Multi-task real robot learning with generalizable
 686 neural feature fields. In *7th Annual Conference on Robot Learning*, 2023.
- 687 Yanjie Ze, Gu Zhang, Kangning Zhang, Chenyuan Hu, Muhan Wang, and Huazhe Xu. 3d diffusion
 688 policy: Generalizable visuomotor policy learning via simple 3d representations, 2024. URL
 689 <https://arxiv.org/abs/2403.03954>.

702 Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language
703 image pre-training, 2023. URL <https://arxiv.org/abs/2303.15343>.

704
705 Tong Zhang, Yingdong Hu, Hanchen Cui, Hang Zhao, and Yang Gao. A universal semantic-
706 geometric representation for robotic manipulation, 2023. URL [https://arxiv.org/abs/](https://arxiv.org/abs/2306.10474)
707 2306.10474.

708 Tong Zhang, Yingdong Hu, Jiacheng You, and Yang Gao. Leveraging locality to boost sample
709 efficiency in robotic manipulation. *arXiv preprint arXiv:2406.10615*, 2024.

710 Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual
711 manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023a.

712 Yan Zhao, Ruihai Wu, Zhehuan Chen, Yourong Zhang, Qingnan Fan, Kaichun Mo, and Hao Dong.
713 Dualafford: Learning collaborative visual affordance for dual-gripper manipulation. In *Internation-
714 al Conference on Learning Representations*, 2023b.

715 Yuchen Zhou, Jiayuan Gu, Xuanlin Li, Minghua Liu, Yunhao Fang, and Hao Su. Partslip++:
716 Enhancing low-shot 3d part segmentation via multi-view instance segmentation and maximum
717 likelihood estimation, 2023. URL <https://arxiv.org/abs/2312.03015>.

718 Haoyi Zhu, Yating Wang, Di Huang, Weicai Ye, Wanli Ouyang, and Tong He. Point cloud matters:
719 Rethinking the impact of different observation spaces on robot learning, 2024. URL <https://arxiv.org/abs/2402.02500>.

720

721

722

723

724

725

726

727

728

729

730

731

732

733

734

735

736

737

738

739

740

741

742

743

744

745

746

747

748

749

750

751

752

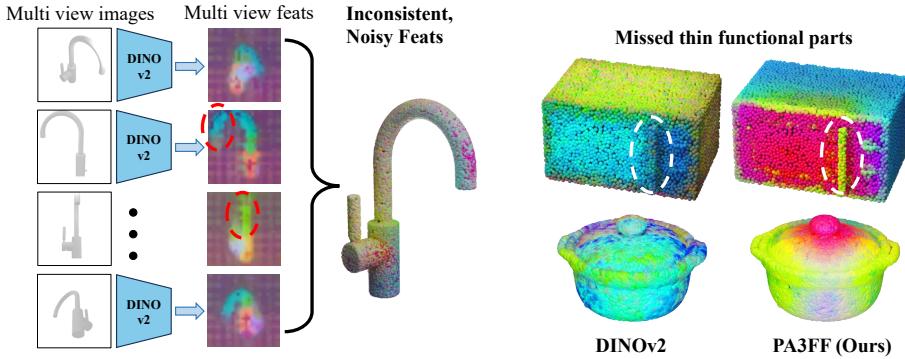
753

754

755

756 A DETAILS OF PA3FF

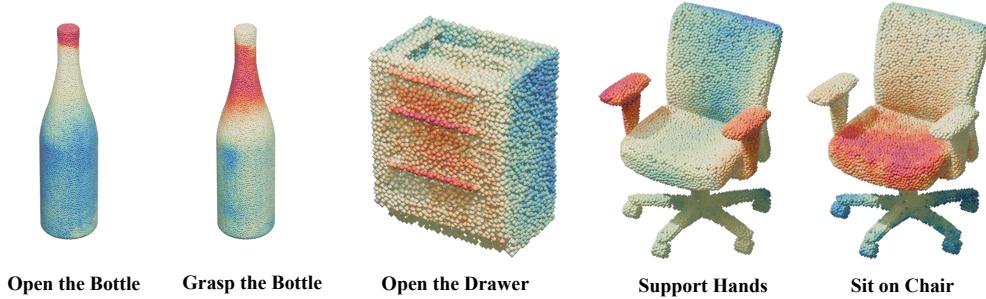
758 A.1 LIMITATIONS OF FEATURE LIFTING UP



773 Figure 7: Flaws of lifting up method.

775 Although 3D priors are known to enhance generalization, lifting features from 2D to 3D introduces
 776 significant challenges. Models that naively average multi-view features from frozen 2D networks
 777 suffer from inconsistent visibility across views. Rendered 2D images can also miss thin or small parts
 778 like handles or buttons.

780 A.2 MORE FEATURES' VISUALIZATIONS



792 Figure 8: Heatmaps of point cloud features' similarity with several queries' text encoding.

795 Since the features generated by PA3FF contain semantic information, calculating feature similarity
 796 using different task statements for the same object allows us to focus on different parts of the object.
 797 The figure shows the heatmap visualization of cosine similarity between different encoded instructions
 798 and features.

800 A.3 MORE QUANTITATIVE RESULTS OF PA3FF

801 Table 4: **Segmentation Results on the PartNetE Dataset.** Category mAP50s (%) are shown for
 802 different object categories. Higher values indicate better performance.

Method	Bottle	Chair	Display	Lamp	Storage Furniture	Table	Average
PointGroup Jiang et al. (2020)	8.0	77.2	16.7	9.8	0.0	0.0	18.6
SoftGroup Vu et al. (2022)	22.4	87.7	27.5	19.4	11.6	14.2	30.5
PartSlip Liu et al. (2023)	79.4	84.4	82.9	68.3	32.8	32.3	63.4
PartSlip++ Zhou et al. (2023)	78.5	86.0	74.1	66.9	36.7	33.5	62.6
Ours	94.6	90.0	86.5	69.5	49.6	33.4	70.6

810 **B ABLATION STUDY**
811

812 To validate the effectiveness of our proposed PADP framework, we conduct comprehensive ablation
813 studies on the individual components of our method. Each ablation targets a specific design choice to
814 understand its contribution to the overall performance.
815

816 **B.1 ABLATION COMPONENTS**
817

818 We evaluate the following key components of our approach:
819

820 **Stacking Additional Transformers:** We modify the PTv3 architecture by removing most downsam-
821 pling layers and instead deepen the network through stacking additional transformer blocks. This
822 architectural change enhances detail preservation while improving feature abstraction capabilities.
823

824 **Feature Refinement via Contrastive Learning:** We apply feature refinement through contrastive
825 learning across different objects to enhance part-level consistency and distinctiveness in the learned
826 representations.
827

828 Table 5: Ablation study results showing the impact of different components on task performance.
829

830

Method	Put in Drawer (%)
PADP (Full Method)	62
w/o Stacking Additional Transformers	58
w/o Feature Refinement	46
Sonata + DP3	39
DP3 (Baseline)	37

837 **B.2 ANALYSIS AND DISCUSSION**
838

839 Table 5 presents the quantitative results of our ablation study on the "Put in Drawer" task, demonstra-
840 ting the contribution of each component to the overall performance. Our ablation study reveals several
841 critical insights into the effectiveness of PADP:
842

843 **Limited Gains from Direct Combination:** The combination of Sonata with DP3 achieves only
844 39% success rate, representing a modest improvement of 2% over the DP3 baseline (37%). This
845 demonstrates that simply integrating existing methods without architectural modifications yields
846 limited performance gains.
847

848 **Impact of Architectural Modifications:** Removing the feature refinement component while main-
849 taining our transformer modifications results in 46% success rate. This 7% improvement over
850 the baseline combination indicates that our architectural changes to the transformer stack provide
851 meaningful benefits for manipulation tasks.
852

853 **Critical Role of Feature Refinement:** The most substantial performance degradation occurs when
854 removing the feature refinement component, dropping from 62% to 46%. This 16% decrease high-
855 lights the importance of contrastive learning for achieving part-level consistency and distinctiveness
856 in the learned representations.
857

858 **Synergistic Effect of Components:** The full PADP method achieves 62% success rate, demonstrating
859 that the combination of architectural modifications and feature refinement creates a synergistic effect
860 that significantly outperforms individual components.
861

862 These findings underscore that PADP's performance gains stem primarily from our novel part-aware
863 feature field learning approach rather than merely utilizing Sonata representations. The substantial
864 improvement from 46% to 62% when including feature refinement confirms that our algorithmic
865 contributions are essential for addressing part-level manipulation challenges, and that Sonata alone is
866 insufficient for solving these complex robotic tasks.
867

864 **C EXPERIMENT SETUP**
 865

866 In this section, we provide a detailed description of the experimental setup, including both real-world
 867 and simulation configurations, as well as a discussion of the baselines.
 868

869 **C.1 REAL-WORLD ENVIRONMENT SETUP**
 870

871 **C.1.1 HARDWARE SETUP**
 872

873 For the real-world experiments, our experimental platform is built around a Franka Emika Panda
 874 robotic arm, with its parallel gripper’s fingers replaced by UMI fingers (Chi et al., 2024b). For
 875 perception, we employ three Intel RealSense D415 depth cameras. Figure 9 shows our real-world
 876 setup and object used.



900 Figure 9: Real-world experiment environment and assets utilized.
 901

902 **C.1.2 TASKS DETAILS.**
 903

904 For the real-world experiments, we selected 8 representative tasks, covering a variety of manipulation
 905 scenarios—including **pulling lid of pot**, **open drawer**, **close box**, **close lid of laptop**, **open**
 906 **microwave**, **open bottle**, **put the lid on the kettle**, and **press dispenser** to evaluate our system’s
 907 performance across diverse challenges, as shown in Figure 3.
 908

- 910 • **Pulling lid of pot**: A pot with a detachable lid is placed on the workspace. The robot must
 911 accurately grasp the lid’s handle, lift it vertically, and separate it from the pot.
- 912 • **Open drawer**: A small drawer unit with a front handle is present. The robot must grasp the
 913 handle and smoothly pull the drawer forward until it is fully open.
- 914 • **Close box**: A small hinged box sits on the table. The robot must grasp the lid’s edge and
 915 rotate it along its hinge axis until the box is fully closed.
- 916 • **Close lid of laptop**: An open laptop lies on the workspace. The robot must grasp the top
 917 edge of the screen and fold it down smoothly until the laptop is fully closed.

- 918
 919
 920
 921
 922
 923
 924
 925
 926
 927
 928
 929
 930
 931
 932
 933
 934
 935
 936
 937
 938
 939
 940
 941
 942
 943
 944
 945
 946
 947
 948
 949
 950
 951
 952
 953
 954
 955
 956
 957
 958
 959
 960
 961
 962
 963
 964
 965
 966
 967
 968
 969
 970
 971
- **Open microwave:** A microwave oven with a front-mounted handle is present. The robot must grasp the handle and pull the door outward until it is fully open.
 - **Open bottle:** A bottle with a screw-on cap stands on the table. The robot must grasp the cap, align its threads with the bottle neck, rotate it in the instructed direction by the required angle until the cap is fully unscrewed, and then lift the cap off vertically.
 - **Put the lid on the kettle:** A kettle with a removable lid is placed on the workspace. The robot must grasp the lid, align it with the kettle opening, place it smoothly on top, and press down vertically until it is securely seated.
 - **Press dispenser:** A dispenser bottle with a squeeze nozzle is placed on the workspace. The robot must grasp the nozzle section with its parallel gripper and apply a squeezing force to actuate the nozzle.

C.2 SIMULATION ENVIRONMENT SETUP

Table 6: Summary of the five test sets and the type of generalization each addresses.

Test Set	Type of Generalization
Test 1 (OS)	Novel object positions and rotations
Test 2 (OI)	Novel object instances within the same category
Test 3 (TP)	Novel part combinations within the same task categories
Test 4 (TC)	Novel part-level manipulation task categories
Test 5 (OC)	Novel object categories

Benchmarks. PartInstruct contains 513 object instances across 14 categories (each annotated with part-level information) and 1302 fine-grained manipulation tasks grouped into 16 task classes. These 16 task classes include 10 seen categories for training and 6 unseen categories for testing, with each category defined by tasks that require the robot to perform a specific combination or sequence of part-level interactions.

Evaluation. To systematically evaluate the performance of the learned policy, PartInstruct designed a five-level evaluation protocol (see Table 6). Each test set evaluates a policy in one type of generalization condition. Specifically, they focus on generalizability over initial **object states (OS)**, novel **object instances (OI)**, novel part combinations in the same **task type (TP)**, novel **task categories (TC)**, and novel **object categories (OC)**. Detailed visualization can be viewed in Figure 10, 11, 12, 13, 14.

550 Move the bottle on the table away from me while holding top, then free it.



Figure 10: Left: Training set. Right: Test 1(OS). Novel object positions and rotations

551 Place gripper tip on the screw of the scissors.



Figure 11: Left: Training set. Right: Test 2(OI). Novel object instances within the same category

972
973
974
975
976
977
978
979
980
981

982  **Grab the top of the mug and move it forwards.**



983
984
985
986
987
988
989
990

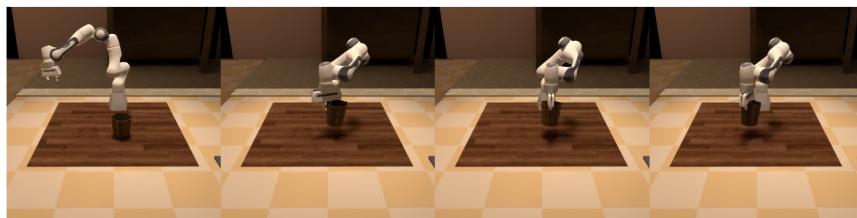
991  **Grab the handle of the mug and move it backwards.**



Figure 12: Above: Training set. Below: Test 3(TP). Novel part combinations within the same task categories

993
994
995
996
997
998
999
1000
1001
1002
1003

1004  **Lift the bucket by its left, then rotate the left part to face front, then move it to the left.**



1005
1006
1007
1008
1009
1010
1011

1012  **Move the bucket towards to the left in the air, then rotate handle to point towards back.**



Figure 13: Above: Training set. Below: Test 4(TC). Novel part-level manipulation task categories

1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025

1026  **Lift the bottle by its top.**



1027  **Lift the knife by its base body.**



Figure 14: Left: Training set. Right: Test 5(OC). Novel object categories

1026
1027

C.3 DETAILS OF BASELINES

1028
1029
1030
1031
1032

Diffusion Policy (DP) We train a CNN-based DP from scratch; the action prediction horizon is set to 16 steps, with an observation horizon of 2 steps and action steps of 8. The input RGB images are cropped to a size of 76×76 . For language instructions, we use a pre-trained T5-small language encoder to obtain a language embedding of 512 dimensions. This language embedding is then concatenated with other features to form the final feature representation.

1033
1034
1035
1036

GenDP Following the raw work (Wang et al., 2024c), we use DINOv2 and Grounding-DINO to extract information from multi-view 2D RGB images, and then project arbitrary 3D coordinates back to each camera, interpolate to compute representations from each view, and fuse these data to derive the descriptors associated with these 3D positions.

1037
1038
1039
1040
1041

3D Diffusion Policy (DP3) The DP3 model is trained under a similar setup as DP, with an action prediction horizon of 16 steps, an observation horizon of 2 steps, and action steps of 8. For the point cloud observations, we use an input size of 1024 points, which are downsampled from the original point cloud using the Iterative Farthest Point Sampling algorithm (Qi et al., 2017). The language instructions are processed in DP3 following the same approach as in DP.

1042
1043
1044
1045
1046

Act3D Act3D takes an image input size of 256×256 . The action prediction horizon is set to 6 steps, and the observation horizon is 1 step. Following the raw work Gervet et al. (2023), we use ResNet50He et al. (2016) as the vision encoder, and use CLIP Radford et al. (2021) embeddings for vision-language alignment. For 3D action map generation, the number of "ghost" points is set to be 10,000, with a number of sampling level of 3.

1047
1048
1049
1050
1051

3D Diffuser Actor (3D-DA) For 3D-DA, we use the front-view RGB and scene point cloud as vision inputs. The RGB image has a resolution of 256×256 . Following Ke et al. (2024a), we extract visual features with a pre-trained CLIP ResNet-50 encoder and use CLIP Radford et al. (2021) embeddings for vision-language alignment. We use an interpolation length of 5 steps and an observation history of 3 steps.

1052
1053
1054
1055
1056

RTV2 We first convert the depth map from the static camera view into a point cloud in the camera coordinates, then apply camera extrinsic to transfer the point cloud into the world coordinates, where the action heat maps will be generated, and apply supervision. The action prediction horizon is chosen to be 6 steps, and the observation horizon is set to be 1 step.

1057
1058

D MORE REAL-WORLD EXPERIMENT RESULTS

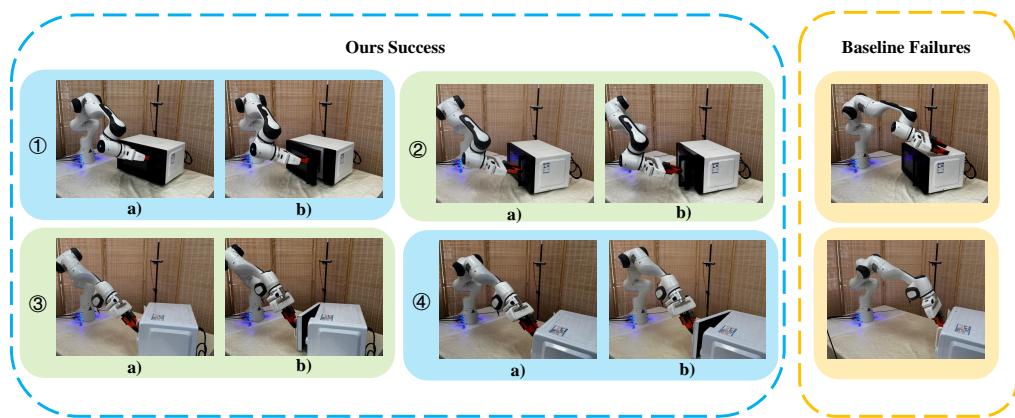
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074

Figure 15: Our policy reliably detects variations in object parts across different positions and successfully executes the corresponding tasks, whereas the baseline method exhibits weaker spatial generalization and struggles to accurately perceive the object parts after positional changes.

1075
1076
1077
1078
1079

1. **Spatial Generalization of PADP:** In the “Open microwave” task, our policy accurately perceives and localizes the door handle even when the microwave is spatially displaced,

1080
 1081
 1082
 1083
 1084
 1085
 1086
 1087
 1088
 1089
 1090
 1091
 1092
 1093
 1094
 1095
 1096
 1097
 1098
 1099
 1100
 1101
 1102
 1103
 1104
 1105
 1106
 1107
 1108
 1109
 1110
 1111
 1112
 1113
 1114
 1115
 1116
 1117
 1118
 1119
 1120
 1121
 1122
 1123
 1124
 1125
 1126
 1127
 1128
 1129
 1130
 1131
 1132
 1133

allowing it to reliably grasp the handle and pull the door open (shown in Figure 15 left). In contrast, the baseline method exhibits weaker spatial generalization: it fails to correctly identify the microwave and its handle after translation or rotation, often grasping incorrect regions and failing to open the door (shown in Figure 15 right).

2. **Environment generalization of PADP:** Other than the “Open microwave” task, we also conducted environment generalization experiments on the “Open bottle” task. As shown in Table 3 and Figure 5, our policy maintains robust performance across varied spatial configurations, whereas baseline methods suffer a significant drop in success rate.

E MORE SIMULATION RESULT

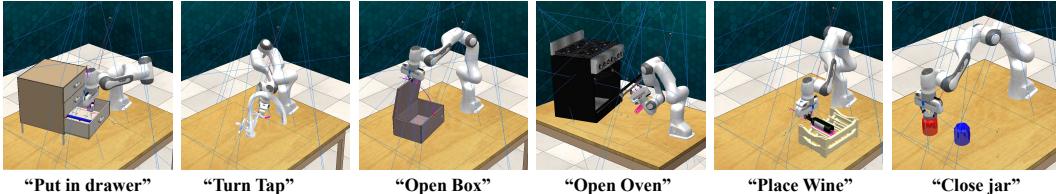


Figure 16: Visualizations of simulation environments.

Benchmarks. RLBench comprises 100 completely unique, hand-designed tasks performed using a 7-DoF Franka Emika Panda arm with a parallel gripper. Observations include rich proprioceptive data (joint angles, velocities, forces, gripper state) and visual data (RGB, depth, segmentation masks) from an over-the-shoulder stereo camera and an eye-in-hand monocular camera. Each task provides an infinite supply of motion-planned demonstrations, supporting research in imitation and few-shot learning.

Tasks Details. For our simulation experiments, we select 6 tasks from the RLBench, as shown in Figure 16.

- **Put item in drawer**: In this scenario, a cabinet features three drawers—upper, middle, and lower—with a small cube resting on top. The objective is for the robot to grasp the bottom drawer’s handle, pull it open to an accessible position, and then precisely insert the cube into the opened drawer.
- **Turn tap**: In this scenario, two taps—one on the left and one on the right—each equipped with a rotary handle, are present. The objective is for the robot to interpret grasp that left tap’s handle, and rotate it in the instructed direction until the tap is fully turned on.
- **Open box**: In this scenario, a box with a hinged lid is placed before the robot. The objective is to accurately grasp the edge of the lid and lift it along its hinge axis until the box is fully open.
- **Open oven**: In this scenario, an oven with a front-facing handle is present. The objective is for the robot to firmly grasp the oven door handle and pull it outward along its hinge until the oven is fully open.
- **Stack wine**: In this scenario, a standard wine bottle and a horizontal-slot wine rack—designed to cradle bottles on their sides—are placed on the work surface. The objective is for the robot to securely grasp the bottle, slide it into the designated rack slot on its side, and ensure it is evenly supported and perfectly centered without tilting.
- **Close jar**: In this scenario, two jars and a single lid are presented. The robot must grasp the lid, align its threads with the jar on the left, and rotate it in the instructed direction until the lid is fully tightened and sealed.

Training Details. We collected expert demonstrations using the built-in scripted policies in RLBench, gathering 25 and 50 demonstrations per task for training. For DP and DP3, after training for at least 2000 epochs, we select the single best-performing epoch for evaluation: we run each policy over 10 episodes, choose the top three success rates, and report their average as the final performance metric.

1134 For our policy, we instead select the best-performing epoch within the range of epochs **300** to **400**
 1135 and apply the same evaluation protocol.

1136
 1137 **Results.** We evaluate our method on 6 RLBench tasks, as shown in Figure 16 and Table 7. Consistent
 1138 with the real-world results, the simulation results also demonstrate that PADP enhances learning
 1139 efficiency and generalization.

1140 Table 7: Simulated results with different numbers of demonstrations. Here we use the two seen
 1141 objects in the training phase to test the success rate, and conduct five trials for each object with
 1142 **random initialization** in each task.

Method/Task Number of Demos	<i>Put in Drawer</i>		<i>Turn Tap</i>		<i>Open Box</i>		<i>Open Oven</i>		<i>Place Wine</i>		<i>Close Jar</i>	
	25	50	25	50	25	50	25	50	25	50	25	50
DP	16%	22%	19%	26%	13%	18%	23%	31%	11%	16%	4%	7%
DP3	31%	37%	41%	47%	34%	39%	51%	59%	41%	49%	9%	13%
GenDP	38%	44%	48%	55%	41%	48%	63%	68%	48%	56%	14%	19%
PADP (Ours)	52%	62%	61%	69%	56%	66%	77%	86%	63%	74%	22%	30%

1143
 1144
 1145
 1146
 1147
 1148 Table 8: Runtime efficiency comparison of different policy learning methods.

Method	Inference Speed (FPS)
Diffusion Policy	12.3
DP3	12.7
GenDP	0.67
PADP (Ours)	4.23

1149 1150 F RUNTIME EFFICIENCY ANALYSIS

1151 To evaluate the computational efficiency of our proposed PADP method, we conduct inference speed
 1152 comparisons with state-of-the-art baselines. All experiments are performed on the same hardware
 1153 configuration to ensure fair comparison. Table 8 presents the inference speed comparison across
 1154 different methods, measured in frames per second (FPS) during policy execution. While our method
 1155 is slower than DP3 approaches due to 3D processing overhead, it significantly outperforms GenDP
 1156 and maintains practical real-time performance for robotic manipulation.

1157 1158 G LIMITATION

1159 Our work focuses primarily on rigid and articulated objects. While it can be extended to handle
 1160 flexible objects with relatively simple deformations—such as cables or ropes—it remains limited
 1161 when applied to deformable objects exhibiting complex structural changes after deformation. The
 1162 foundation model we rely on, Sonata, struggles to cope with such challenging cases. Addressing
 1163 objects with complex deformations remains an open problem for future research. We hope our work
 1164 can inspire further exploration in this direction.

1165 1166 H THE USE OF LARGE LANGUAGE MODELS

1167 We used a Large Language Model (LLM) only as a writing assistant to polish the language of the
 1168 manuscript (*e.g.*, grammar refinement, style adjustment, and clarity improvement). The research
 1169 ideas, methodology design, experiments, and analysis were entirely conceived, implemented, and
 1170 validated by the authors without reliance on the LLM. The LLM did not contribute to research
 1171 ideation, experimental design, or result interpretation.