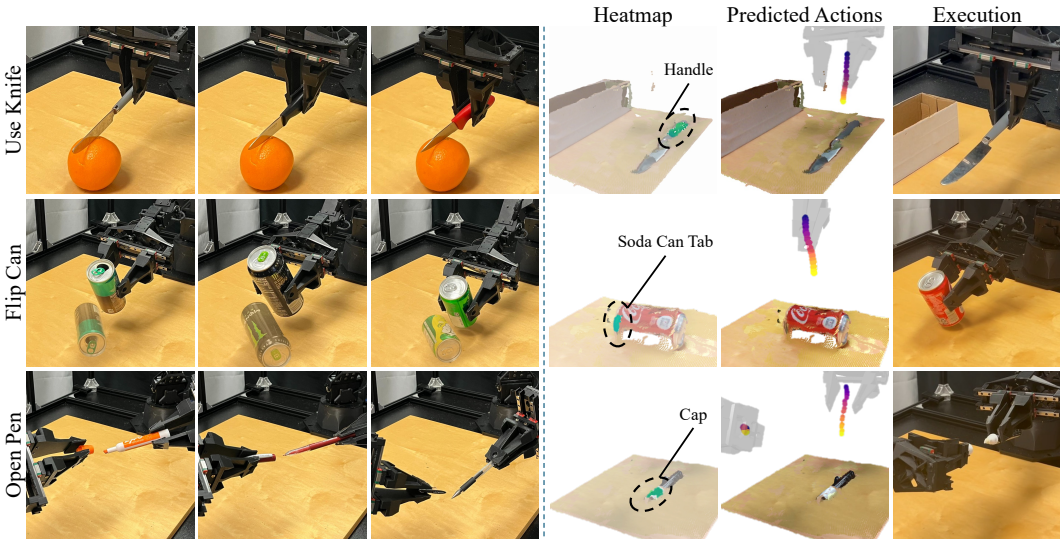
GenDP: 3D Semantic Fields for Category-Level Generalizable Diffusion Policy

Yixuan Wang[1], **Guang Yin**[2], **Binghao Huang**[1], **Tarik Kelestemur**[3], **Jiuguang Wang**[3], **Yunzhu Li**[1]

[1]Columbia University, [2]University of Illinois, Urbana-Champaign, [3]Boston Dynamics AI Institute

Abstract: Diffusion-based policies have shown remarkable capability in executing complex robotic manipulation tasks but lack explicit characterization of geometry and semantics, which often limits their ability to generalize to unseen objects and layouts. To enhance the generalization capabilities of Diffusion Policy, we introduce a novel framework that incorporates explicit spatial and semantic information via 3D semantic fields. We generate 3D descriptor fields from multi-view RGBD observations with large foundational vision models, then compare these descriptor fields against reference descriptors to obtain semantic fields. The proposed method explicitly considers geometry and semantics, enabling strong generalization capabilities in tasks requiring category-level generalization, resolving geometric ambiguities, and attention to subtle geometric details. We evaluate our method across eight tasks involving articulated objects and instances with varying shapes and textures from multiple object categories. Our method demonstrates its effectiveness by increasing Diffusion Policy's average success rate on *unseen* instances from 20% to 93%. Additionally, we provide a detailed analysis and visualization to interpret the sources of performance gain and explain how our method can generalize to novel instances. Project page

Keywords: Semantic Fields, Category-Level Generalization, Imitation Learning, Diffusion Models



(a) Daily Tasks Examples (b) Our Imitation Learning Framework

Figure 1: **Generalizable Diffusion Policy using 3D Semantic Fields.** Our approach introduces a diffusion policy capable of generalizing to new instances within a category by utilizing 3D semantic fields. These fields distinguish semantically meaningful parts of objects in 3D space, as illustrated in the heatmap example. Panel (a) on the left showcases daily task examples where semantic understanding is crucial, while panel (b) on the right demonstrates our method's ability to highlight semantically meaningful parts, such as a knife handle, and how our predicted policy accomplishes these tasks using the 3D semantic fields.

8th Conference on Robot Learning (CoRL 2024), Munich, Germany.

1 Introduction

Diffusion-based policies have recently shown promising results in imitation learning for real robot deployment in complex robotic manipulation tasks [1, 2], because they can express multimodal action distribution, output high-dimensional actions, and train stably. However, existing end-to-end diffusion policy frameworks do not generalize to novel instances due to their brittleness to environment variances, such as object appearances and background changes. Our work proposes a diffusion policy framework capable of category-level generalization and robust to environmental changes using 3D semantic fields.

Previous work has explored the use of point clouds as policy inputs to help generalization [2, 3], but the reliance on geometric information is insufficient for fine-grained scene understanding and generalization. For instance, as illustrated in Figure 1 (b), a marker's head and tail are geometrically ambiguous despite their functional and semantic differences. In this work, we augment geometric information with semantics to generalize to novel instances, resolve geometric ambiguity, and attend to subtle geometric details.

An ideal representation should not only extract geometric information from raw observation but also retain semantic information for better category-level generalization. In this work, we introduce a diffusion policy framework that uses a scene representation in the form of **3D semantic fields**. The 3D semantic fields are defined as the similarity fields, where the similarity score is higher for points closer to one object part, such as knife handles. Note that the 3D semantic fields have multiple channels, as shown in Figure 2, where the first channel corresponds to the shoe tongue, and the last one corresponds to the shoe cuff. Our framework consists of three main modules: a 3D descriptor fields encoder, a semantic fields constructor, and an action policy. The 3D descriptor encoder takes in multi-view RGBD observations and outputs a point cloud with high-dimensional descriptors using large foundational vision models like DINOv2 [4]. These descriptors are then fed into the semantic fields constructor and converted into low-dimensional semantic fields. Finally, the policy takes in the semantic fields along with the point cloud as inputs and predicts actions.

The proposed framework offers three benefits: (1) **Category-level generalization:** As semantic fields contain both 3D and semantic information, it guides our policy to focus on semantically meaningful parts essential for task completion, allowing generalization across instances within a category. (2) **Resolve geometric ambiguity:** Geometric information can be ambiguous. For example, the knife blade and knife handle are geometrically similar despite functional and semantical differences. Our semantic fields can localize space semantically close to parts important for task completion, such as knife blades, to disambiguate vague geometric information. (3) **Attention to subtle semantic details**: Semantic details might be lost due to real-world observation noise, such as toothbrush head and soda can tab. Some tasks are impossible to accomplish without sufficient details, such as spreading toothpaste on a toothbrush and flipping a lying soda can upwards, as shown in Figure 1. Because our semantic fields highlight semantically distinct regions, our method can pay attention to nuanced semantic details for task completion.

We systematically evaluate our method across eight tasks. Our task settings involve novel instances, ambiguous object geometry, and require attention to subtle semantic details. Compared to baseline methods, which frequently fail to generalize to novel instances due to a lack of both 3D and semantic information in their representation, our approach demonstrates significant advantages in category-level generalization. It also leads to substantial performance improvements in scenarios containing geometric ambiguity and subtle details. Through comprehensive experiments, we show an improvement in Diffusion Policy's average success rate on *unseen* instances from 20% to 93%. We also provide a detailed analysis of how well and why our method can generalize to novel instances.

In this work, our contributions are threefold: (1) We propose a diffusion policy framework that uses 3D semantic fields, which leverages visual foundation models to encode the environment's geometric and semantic information. (2) Our design allows category-level generalization, differentiating geometric ambiguity, and attending to geometric details. (3) We conduct comprehensive experiments and improve the original diffusion policy's average success rate on *unseen* instances from 20% to 93%. We also provide a detailed analysis of how well and why our method can generalize to novel instances.

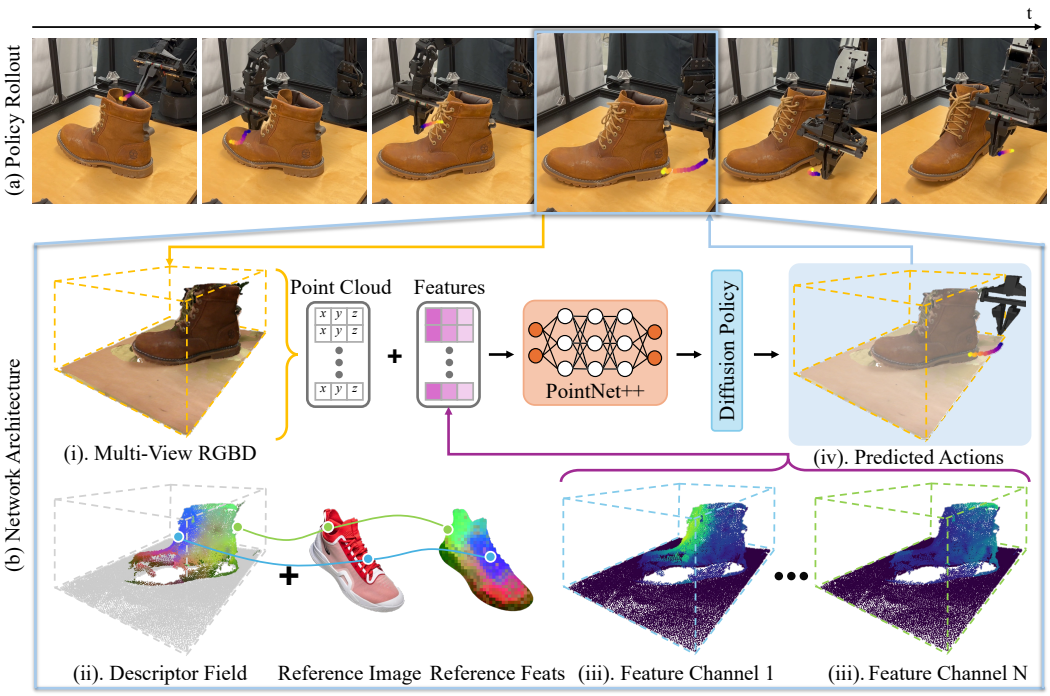


Figure 2: **Method Overview.** The top row (a) shows a sequence of real policy rollouts in the aligning shoe task. We first take in multi-view RGBD observations (i), then extract the 3D descriptor field, with each point possessing a corresponding high-dimensional descriptor (ii) [5]. We then select reference features from 2D reference images. By computing the cosine similarity between the descriptor field and 2D reference semantic features, we could obtain several semantic fields (iii). These semantic fields, concatenated with the point cloud, are then input into PointNet++ and the diffusion policy to output predicted actions (iv).

2 Related Works

Diffusion Models for Robotics. Beginning with random samples, diffusion models operate by iterative denoising to reveal the underlying distribution. They are widely used in robotics, including navigation [6], manipulation [1, 2, 7–14], control [15], and planning [16–18]. Among these works, Diffusion Policy and 3D Diffusion Policy are closely related to our work [1, 2].

Diffusion Policy produces a visuomotor policy under the action diffusion model, which can model the multi-modal distribution of demonstration actions in the real world. However, it relies on multi-view RGB observations, which are brittle to environment changes, including novel object instances, camera viewpoints, and background changes, and fail to generalize to novel instances. Instead of RGB observations, our work enables category-level generalization using 3D semantic fields.

3D Diffusion Policy approaches this problem using point clouds as inputs [2]. While this is proven to be sample-efficient, point clouds without semantic information can be insufficient for many manipulation tasks. In this work, we augment geometric information with semantic information using foundational vision models.

Geometric Representation in Robotic Manipulation. One typical geometric representation used in robotics is point clouds [3, 19–26], which enables the policy to focus on geometry and facilitates generalization across different environments. Yet, this often results in the loss of critical RGB data, limiting the understanding of an object's semantic properties. Unlike these approaches, our approach retains both semantic and geometric information, enhancing downstream decision-making.

Another line of works uses keypoints as the geometric representation in robotic manipulation [27–34]. They have shown impressive generalization capabilities and high efficiency due to their low dimension. However, the resulting sparsity loses geometric details and limits the range of tasks that can be effectively addressed. In contrast, our framework employs a representation with rich geometric information and enables a broader range of tasks.

Semantic Representation in Robotic Manipulation. Semantic reasoning is an ongoing field in robotics [35]. One common semantic representation learning frame is directly learning object-centric or scene-centric embeddings from RGB observations [36–44]. However, these representations are hard to generalize to new environments with different backgrounds and lighting conditions. In contrast, we use 3D semantic representation, which is generalizable across diverse environments and instances. Another class of semantic representation extracts functional or affordance information from observations [43, 45–52]. This line of work focuses on tasks that can be accomplished using motion primitives such as grasping, picking, and placing, while our diffusion-based policy is more flexible in action representation and tasks.

A recent line of research used neural implicit models such as Occupancy Networks [53] or NeRFs [54] to encode semantic features [5, 9, 25, 55–61]. GeFF is the closest one to our work. It operates by utilizing FeatureNeRF [62], which distills a NeRF model from foundational models. Although the distillation process allows for building NeRF upon sparse input views, it reduces the generalization capability. In contrast, we extract 3D descriptor fields without distillation, which inherits generalization capabilities from foundational vision models. In addition, GeFF uses predefined open-push-close action sequences to solve manipulation tasks. Our method differs by deploying an imitation learning policy that could accomplish more complicated tasks.

3 Method

3.1 Problem Statement

We define our system as a Markov Decision Process (MDP) consisting of state $s \in \mathcal{S}$ and action $a \in \mathcal{A}$. The system transition is defined through the dynamics model $s_{t+1} = f(s_t, a_t)$. We assume predicted actions follow the probability distribution $a_t \sim \mathcal{P}_{\text{pred}}(a|s_t) = \pi(s_t)$, where π is the learned policy. The goal is to minimize the difference between $\mathcal{P}_{\text{pred}}(a|s_t)$ and ground truth action probability $\mathcal{P}_{\text{gt}}(a|s_t)$, given as a set of human demonstrations $D = \{\tau_0, \tau_1, ..., \tau_N\}$, where τ_i represents a trajectory comprising $\{s_0, a_0, s_1, ..., a_T\}$. Here, the state s consists of a sequence of multi-view RGBD observations, and the action a is a sequence of robot states.

3.2 3D Descriptor Fields

We use an off-the-shelf large foundational vision model, DINOv2 [4], to obtain semantic features from multi-view RGBD observations. Given its ability to extract consistent semantic features from the RGB images across context and instance variances, we selected it as the backbone network.

We provide pseudocode for building 3D descriptor fields in Algorithm 1. We denote single-view RGBD observation as $\boldsymbol{o}_i = (\mathcal{I}_i, \mathcal{R}_i)$, with $i \in \{1, 2, ..., N\}$ representing the camera index, consisting of an RGB image $\mathcal{I}_i \in \mathbb{R}^{H \times W \times 3}$ and a depth image $\mathcal{R}_i \in \mathbb{R}^{H \times W}$. We first use DINOv2 to extract dense 2D feature maps $\mathcal{W}_i$ corresponding to the RGB image $\mathcal{I}_i$ [4]. For an arbitrary 3D point p, we project it onto the image space to find its corresponding pixel location u_i and the distance to camera r_i. We then interpolate to derive features f_i from the feature map and the depth r'_i from $\mathcal{R}_i$.

The depth difference $\Delta r_i = r'_i - r_i$ reflects how distant p is from the surface. When p is closer to the surface in view i, greater weight is given to f_i. We fuse features from multiple viewpoints by applying a weighted sum, thus obtaining the descriptor f corresponding to p. In practice, we follow the implementation details in D^{3}Fields [5] to extract point cloud $\mathcal{P} \in \mathbb{R}^{K \times 3}$ and associated features $\mathcal{F} \in \mathbb{R}^{K \times F}$, where K is the point cloud's size.

3.3 3D Sematic Fields

First, we obtain several 2D images containing the task-relevant object category and extract feature maps. Then we select features from each feature map and take the average to get a set of reference descriptors $\mathcal{F}_{\text{ref}} \in \mathbb{R}^{M \times F}$. Each descriptor represents a part of the object, such as the shoe head.

We denote the set of reference descriptors $\mathcal{F}_{\text{ref}}$ with M descriptors and descriptor fields $\mathcal{F}$ with K descriptors such that $\mathcal{F} = \{\mathcal{F}_i \in \mathbb{R}^F | i \in \{1, ..., K\}\}$ and $\mathcal{F}_{\text{ref}} = \{\mathcal{F}_{\text{ref},j} \in \mathbb{R}^F | j \in \{1, ..., M\}\}$. The semantic fields $\mathcal{C} \in \mathbb{R}^{K \times M}$ are defined as the similarity between the descriptor fields and reference descriptors:

$$\mathcal{C}_{ij} = \frac{\mathcal{F}_i \cdot \mathcal{F}_{\text{ref},j}}{||\mathcal{F}_i|| ||\mathcal{F}_{\text{ref},j}||}. \tag{1}$$

By computing the similarity scores, we convert high-dimensional descriptor fields into M-dimensional semantic fields, where M is typically less than 5. We then concatenate semantic fields $\mathcal{C}$ and raw point cloud $\mathcal{P}$ together, which are then inputted into the policy.

3.4 Policy Learning

In this work, we model our policy as Denoising Diffusion Probabilistic Models (DDPMs), similar to Diffusion Policy [1, 63]. Instead of regressing the action directly, we train a noise predictor network

$$\widehat{\epsilon^k} = \epsilon_\theta(a^k, s, k), \tag{2}$$

Algorithm 1 3D Descriptor Fields Computation

Infer feature map $\mathcal{W}_i$ from single RGB image $\mathcal{I}_i$
procedure EVALUATE(p)
  Project p to camera i and compute projected pixel u_i and distance to camera r_i
  Obtain interpolated features $f_i = \mathcal{W}_i[u_i]$
  Obtain depth $r'_i = \mathcal{R}_i[u_i]$
  Compute depth difference $\Delta r_i = r'_i - r_i$
  Fuse features $\mathcal{F}_j = h(f_i, \Delta r_i), i \in \{1, 2, ..., N\}$

that takes in noisy actions a^k, current observations s, and denoising iterations k and predicts the noise $\widehat{\epsilon^k}$. During training, we randomly choose a denoising step k and sample noise ϵ^k added to the unmodified sample a^0. Our training loss is the difference between ϵ^k and predicted noise:

$$\mathcal{L} = \text{MSELoss}(\epsilon^k, \widehat{\epsilon^k}). \tag{3}$$

During the inference time, our policy starts from random actions a^K and denoises for K steps to obtain the final action predictions. At each step, the action is updated following

$$a^{k-1} = \alpha\big(a^k - \gamma\epsilon_\theta(a^k, s, k) + \mathcal{N}(0, \sigma^2 I)\big), \tag{4}$$

where α, γ, and σ are hyperparameters. In practice, we input 3D semantic fields into PointNet++ and obtain visual features. Then we feed visual features into diffusion policy, similar to [1].

(a) Robot Setup (b) Objects

Figure 3: **Real Experiment Setup.** (a) We use four RealSense cameras to capture RGBD observations and ALOHA robots to execute policy. (b) We test on a diverse set of objects, including shoes, soda cans, marker pens, knives, spoons, toothbrushes, and toothpaste, with diverse geometry and appearance.

4 Experiments

In this section, we evaluate our method on eight diverse tasks. We aim to answer the following three questions through experiments. 1) How does the performance of our method compare to that of the state-of-the-art imitation learning methods? 2) How well can our method generalize to different configurations and instances? 3) What enabled our method to generalize to novel instances?

4.1 Setup

For experiments in simulation, we use SAPIEN [64] to build the environments and evaluate the policies. For the real-world experiments, we use the ALOHA robot and four RealSense cameras for real-world data collection and testing [65]. Figure 3 shows our real-world setup and objects used.

We evaluate our method on eight tasks, as shown in Figure 4. We test in the simulation on **Hang Mug** and **Insert Pencil** tasks. In the real world, we evaluate our method's ability from different perspectives, such as insignificant geometric details, category generalization, and geometric ambiguity. We collected 200 demonstration episodes for the **Hang Mug** task, 100 episodes for **Insert Pencil** task, and 60 episodes for all real-world tasks. More details are listed in the supplementary material. For each task, we train one policy to accomplish that single task.

We test our method on various object categories, including shoes, toothbrushes, soda cans, and knives, among others, which present significant challenges for category-level generalization due to

factors such as large geometric variances, ambiguous geometric information, and nuanced geometric details. More detailed descriptions of the tasks can be found in the supplementary material.

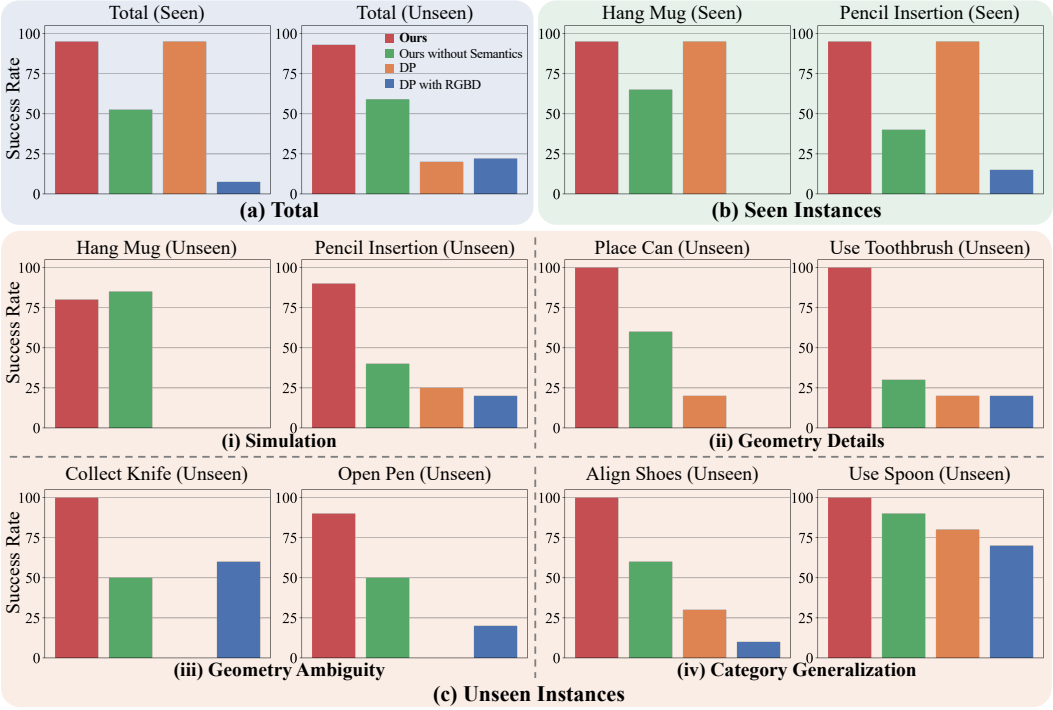


Figure 4: **Success Rate.** Our method was evaluated across eight tasks. (a) The aggregated quantitative results show that our method has similar performance as diffusion policy for seen instances, but our method outperforms all baselines on unseen instances. (b) For the seen instances, diffusion policy and our method have similar performances on two simulation tasks. (c) Diffusion policy performance degrades significantly on unseen instances. In addition, our method outperforms all other baselines, which underscores our policy's capability to attend to geometric details, distinguish geometric ambiguities, and generalize to novel instances.

4.2 Comparison with Baselines

We compare our method with three baselines: 1) **Ours without Semantics**: input raw point cloud without semantic fields, 2) vanilla **Diffusion Policy (DP)**, and 3) **DP with RGBD**: vanilla DP but with depth observations. For all baselines, we utilize four cameras to obtain observations and input them into the policy. We evaluate different policies by the success rate. The quantitative result is summarized in Figure 4 with detailed numbers in the supplementary material.

Our method shows similar performance with the original diffusion policy in the simulation on seen instances. However, when generalizing to unseen instances, diffusion policy performance degrades significantly, because it takes RGB images as input, which is brittle to environmental factors like object appearances and geometries. Therefore, the diffusion policy predicts undesirable actions and fails to accomplish the task. In contrast, our framework encodes explicit 3D and semantic information about the scene, which is robust to object appearance and geometry variances, which helps the policy to generalize to novel instances.

In addition, our method consistently outperforms ours *without* semantics, whether for seen instances or unseen instances. The benefits from 3D semantic fields are threefold. First, they help to attend to geometric details, such as the can tab, to place the can on the table upright, while ours without semantics cannot when solely relying on geometry information. Second, 3D semantic fields help to distinguish geometric ambiguity. For instance, when grasping the knife, it is hard to tell the knife directions from ambiguous geometry, which might lead to unsafe actions like grasping the knife blade. Our 3D semantic fields encode semantic information like knife handles to distinguish geometric ambiguities and show a higher success rate in Figure 4 (c)(iii). Third, Figure 4 (c)(iv) demonstrates our 3D semantic fields could help policy focus on semantically meaningful parts to complete tasks and generalize to novel instances.

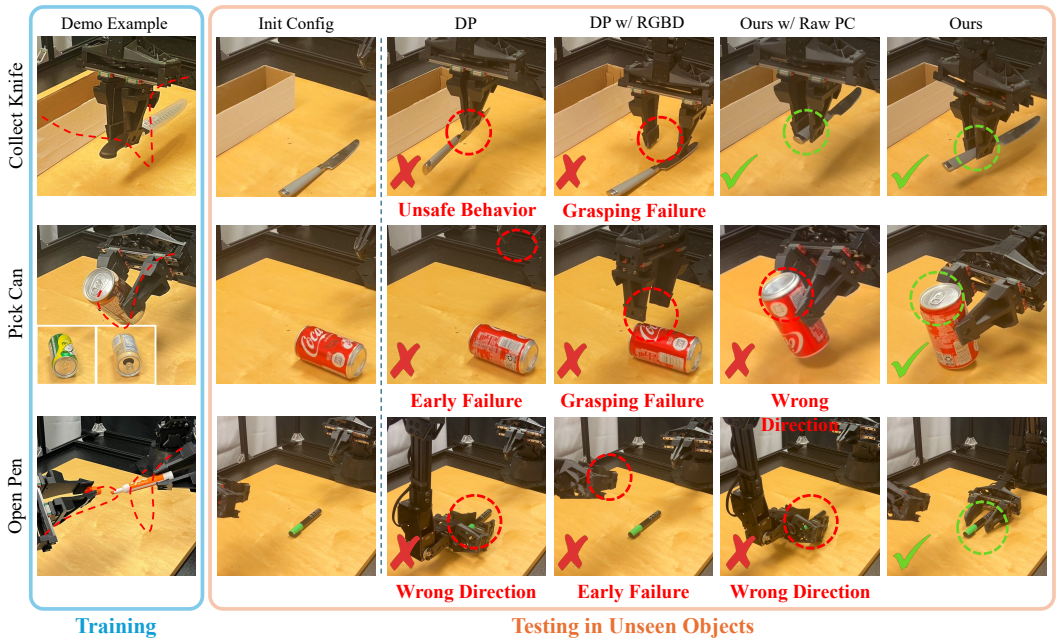


Figure 5: **Policy Rollout in Real World.** The figure illustrates the policy rollout results in the real world. On the left, the blue block displays the demonstration examples and corresponding training instances. On the right, the orange block presents policy rollout results. From left to right, they are, respectively, initial configurations, diffusion policy, diffusion policy with RGBD, ours *without* semantics, and our method. We summarize four common failure modes. Early failure and grasping failure could happen when the novel instance is presented. Diffusion policy may also lead to unsafe behavior when encountering novel instances. Ours *without* semantics might identify wrong directions due to geometric ambiguity and nuanced geometric details.

We found that Diffusion Policy with RGBD is not as effective as our method. Directly adding depth observation to the diffusion policy will make the input space even larger. Therefore, it would require more demonstrations to have sufficient training data coverage, which can make it less data efficient.

Figure 5 shows real-world policy rollout results. We notice different failure modes for baseline methods. For diffusion policy and diffusion policy with RGBD, when a novel instance is presented, they may stop early and fail to make progress. Due to their reliance on 2D observations, they cannot generalize to unseen instances with varying appearances and geometries. Ours *without* semantics often fail to differentiate geometric ambiguity, like knife handles, while our method could identify the right part for manipulation. Our method also focuses on geometric details, like the soda can tab to accomplish the can flipping task, while ours *without* semantics places the soda can upside down.

4.3 Generalization to Novel Configurations and Instances

We also analyze how well our method can generalize to novel configurations and instances. Figure 6 illustrates the predicted actions of different policies under the same observations. For the seen instances, as shown in the leftmost column, we can see that the predicted trajectories for all policies in the first example point towards the knife handle correctly. However, when the knife direction changes, our method *without* semantics fails to respond accordingly, while the diffusion policy and our method predict actions accordingly. This demonstrates that our method can understand the semantic difference between knife handle and knife blade, which results in safe robot behaviors.

When the novel instance is presented with similar configurations, our method will predict a successful trajectory guiding toward the knife handle, while diffusion policy will predict a nonsmooth trajectory, which shows that it is brittle to environmental factors and lacks category-level generalization capabilities. Although our method *without* semantics predicts a smooth trajectory, it approaches the knife blade, which is unsafe.

4.4 Generalization Analysis

We visualize the 3D semantic fields in Figure 7 by overlaying them with the raw point cloud. These semantic fields benefit the policy for two primary reasons: 1) Highlighted points are semantically meaningful and crucial for task completion. For example, to collect books on the shelf upright, the

robot must recognize the book titles. Geometric information alone is inadequate for task completion. 2) Semantic fields are consistent across different instances. The mug example shows that the activation on the handle is consistent across mugs with various appearances and poses. The category-level consistency enables our method to achieve category-level generalization. We also present t-SNE analysis in the supplementary material.

4.5 Extension to Other IL Methods

We also extended our 3D semantic fields and showed benefits to other imitation learning (IL) methods, such as Action Chunking with Transformer (ACT) [65]. The detailed results are presented in the supplementary material.

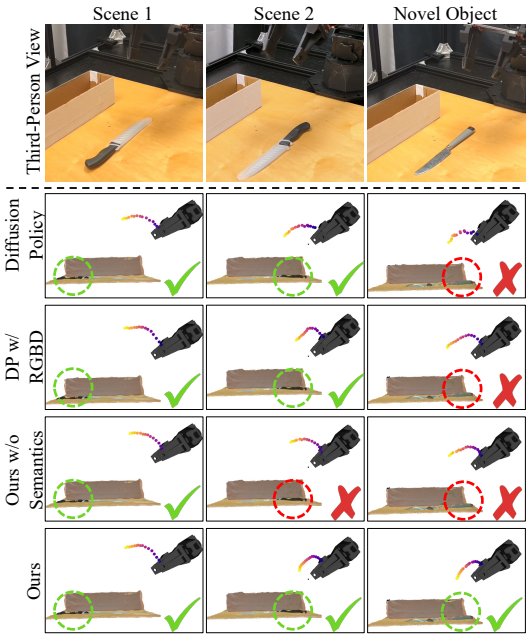


Figure 6: **Predicted Action Comparisons.** We compare different policies' responses under the same state. Diffusion policy's predicted actions change with knife direction accordingly but are not smooth for novel instances. Due to geometric ambiguity, our method *without* semantics fails to distinguish knife directions, and predicts similar actions even under different knife directions, while our method consistently recognizes the knife handle position and predicts correct actions.

5 Conclusion

In this work, we proposed a diffusion policy framework leveraging 3D semantic representations, with policies modeled as DDPMs conditioned on the semantic fields alongside point clouds. We conduct comprehensive physical experiments to show that our framework can generalize to novel instances within the category, resolve geometric ambiguity, and resolve nuanced geometric details. Compared with the vanilla diffusion policy with 20% success rate, our method can reach 93% success rate on unseen instances, demonstrating the effectiveness of our method.

Limitation. For each task, we select a set of fixed reference features to construct semantic features. However, for long-horizon and fine-grained tasks, such as making coffee and assembling, the policy's attention needs to adapt as tasks progress. Constructing task-specific semantic fields that can adaptively select the abstraction level, can be a future direction. In addition, our method could become more interpretable and potentially more efficient if we can incorporate the geometric properties in a more explicit fashion.

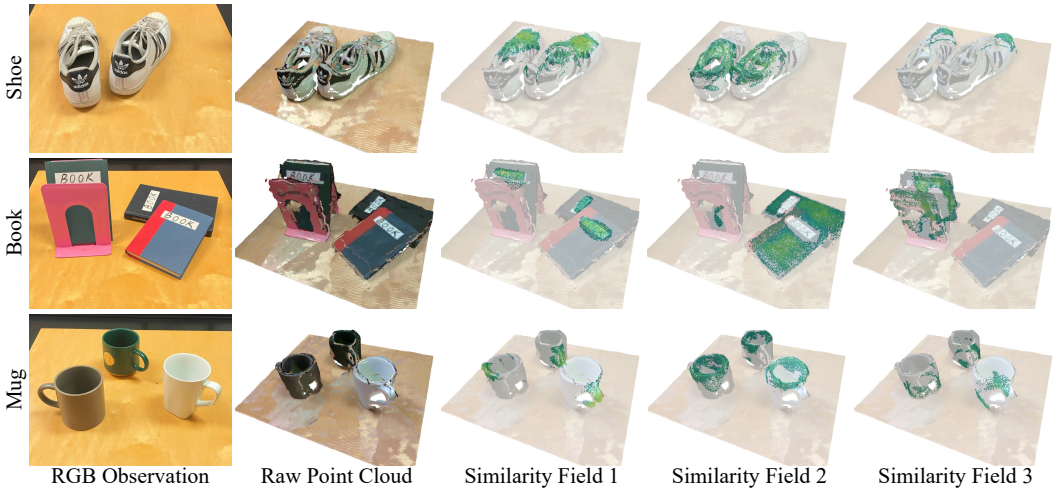


Figure 7: **3D Semantic Fields Visualization.** We visualize 3D semantic fields in the scene with multiple instances from the same category. 3D semantic fields exhibit similar patterns across different instances, such as highlighted book titles of all books. Furthermore, these highlighted areas represent semantically meaningful parts and are important for various tasks, such as shoelaces, shoe heads, book stands, mug handles, and so on.

Acknowledgement

This work is partially supported by Sony. The opinions and conclusions expressed in this article are solely those of the authors and do not reflect those of any Sony entity.

References

[1] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Proceedings of Robotics: Science and Systems (RSS)*, 2023.

[2] Y. Ze, G. Zhang, K. Zhang, C. Hu, M. Wang, and H. Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. *arXiv preprint arXiv:2403.03954*, 2024.

[3] Y. Zhu, Z. Jiang, P. Stone, and Y. Zhu. Learning generalizable manipulation policies with object-centric 3d representations. In *7th Annual Conference on Robot Learning*, 2023.

[4] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, R. Howes, P.-Y. Huang, H. Xu, V. Sharma, S.-W. Li, W. Galuba, M. Rabbat, M. Assran, N. Ballas, G. Synnaeve, I. Misra, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.

[5] Y. Wang, Z. Li, M. Zhang, K. Driggs-Campbell, J. Wu, L. Fei-Fei, and Y. Li. D^3fields: Dynamic 3d descriptor fields for zero-shot generalizable robotic manipulation. *arXiv preprint arXiv:2309.16118*, 2023.

[6] A. Sridhar, D. Shah, C. Glossop, and S. Levine. NoMaD: Goal Masked Diffusion Policies for Navigation and Exploration. *arXiv pre-print*, 2023. URL https://arxiv.org/abs/2310.07896.

[7] Octo Model Team, D. Ghosh, H. Walke, K. Pertsch, K. Black, O. Mees, S. Dasari, J. Hejna, C. Xu, J. Luo, T. Kreiman, Y. Tan, D. Sadigh, C. Finn, and S. Levine. Octo: An open-source generalist robot policy. https://octo-models.github.io, 2023.

[8] T. Pearce, T. Rashid, A. Kanervisto, D. Bignell, M. Sun, R. Georgescu, S. V. Macua, S. Z. Tan, I. Momennejad, K. Hofmann, et al. Imitating human behaviour with diffusion models. *arXiv preprint arXiv:2301.10677*, 2023.

[9] T.-W. Ke, N. Gkanatsios, and K. Fragkiadaki. 3d diffuser actor: Policy diffusion with 3d scene representations. *Arxiv*, 2024.

[10] M. Reuss, M. Li, X. Jia, and R. Lioutikov. Goal conditioned imitation learning using score-based diffusion policies. In *Robotics: Science and Systems*, 2023.

[11] J. Urain, N. Funk, J. Peters, and G. Chalvatzaki. Se(3)-diffusionfields: Learning smooth cost functions for joint grasp and motion optimization through diffusion. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5923–5930, 2023. doi:10.1109/ICRA48891.2023.10161569.

[12] W. Liu, Y. Du, T. Hermans, S. Chernova, and C. Paxton. Structdiffusion: Language-guided creation of physically-valid structures using unseen objects. In *RSS 2023*, 2023.

[13] H. R. P. D. f. M.-m. R. Shelving, Stacking. Sornet: Spatial object-centric representations for sequential manipulation. In *7th Annual Conference on Robot Learning*, 2023.

[14] Z. Xian, N. Gkanatsios, T. Gervet, T.-W. Ke, and K. Fragkiadaki. Chaineddiffuser: Unifying trajectory diffusion and keypose prediction for robotic manipulation. In *7th Annual Conference on Robot Learning*, 2023.

[15] T. Yoneda, L. Sun, G. Yang, B. C. Stadie, and M. R. Walter. To the noise and back: Diffusion for shared autonomy. In *Robotics: Science and Systems XIX, Daegu, Republic of Korea, July 10-14, 2023*, 2023.

[16] Z. Wang, T. Oba, T. Yoneda, R. Shen, M. Walter, and B. C. Stadie. Cold diffusion on the replay buffer: Learning to plan from known good states. In *7th Annual Conference on Robot Learning*, 2023. URL https://openreview.net/forum?id=AyRr_i028w.

[17] K. Saha, V. Mandadi, J. Reddy, A. Srikanth, A. Agarwal, B. Sen, A. Singh, and M. Krishna. Edmp: Ensemble-of-costs-guided diffusion for motion planning, 2023.

[18] M. Janner, Y. Du, J. Tenenbaum, and S. Levine. Planning with diffusion for flexible behavior synthesis. In *International Conference on Machine Learning*, 2022.

[19] A. Goyal, J. Xu, Y. Guo, V. Blukis, Y.-W. Chao, and D. Fox. Rvt: Robotic view transformer for 3d object manipulation. *arXiv preprint arXiv:2306.14896*, 2023.

[20] S. James, K. Wada, T. Laidlow, and A. J. Davison. Coarse-to-fine q-attention: Efficient learning for visual robotic manipulation via discretisation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13739–13748, 2022.

[21] T. Gervet, Z. Xian, N. Gkanatsios, and K. Fragkiadaki. Act3d: 3d feature field transformers for multi-task robotic manipulation. In *Conference on Robot Learning*, pages 3949–3965. PMLR, 2023.

[22] Y. Qin, B. Huang, Z.-H. Yin, H. Su, and X. Wang. Dexpoint: Generalizable point cloud reinforcement learning for sim-to-real dexterous manipulation. In *Conference on Robot Learning*, pages 594–605. PMLR, 2023.

[23] D. Bauer, T. Patten, and M. Vincze. Reagent: Point cloud registration using imitation and reinforcement learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14586–14594, 2021.

[24] T. Chen, M. Tippur, S. Wu, V. Kumar, E. Adelson, and P. Agrawal. Visual dexterity: In-hand reorientation of novel and complex object shapes. *Science Robotics*, 8(84):eadc9244, 2023. doi:10.1126/scirobotics.adc9244. URL https://www.science.org/doi/abs/10.1126/scirobotics.adc9244.

[25] A. Mousavian, C. Eppner, and D. Fox. 6-dof graspnet: Variational grasp generation for object manipulation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2901–2910, 2019.

[26] Y. Yuan, H. Che, Y. Qin, B. Huang, Z.-H. Yin, K.-W. Lee, Y. Wu, S.-C. Lim, and X. Wang. Robot synesthesia: In-hand manipulation with visuotactile sensing. *arXiv preprint arXiv:2312.01853*, 2023.

[27] Y. Li, J. Wu, R. Tedrake, J. B. Tenenbaum, and A. Torralba. Learning particle dynamics for manipulating rigid bodies, deformable objects, and fluids. In *ICLR*, 2019.

[28] L. Manuelli, Y. Li, P. Florence, and R. Tedrake. Keypoints into the future: Self-supervised correspondence in model-based reinforcement learning. *arXiv preprint arXiv:2009.05085*, 2020.

[29] H. Shi, H. Xu, Z. Huang, Y. Li, and J. Wu. Robocraft: Learning to see, simulate, and shape elasto-plastic objects with graph networks. *arXiv preprint arXiv:2205.02909*, 2022.

[30] H. Shi, H. Xu, S. Clarke, Y. Li, and J. Wu. Robocook: Long-horizon elasto-plastic object manipulation with diverse tools. *arXiv preprint arXiv:2306.14447*, 2023.

[31] Y. Wang, Y. Li, K. Driggs-Campbell, L. Fei-Fei, and J. Wu. Dynamic-Resolution Model Learning for Object Pile Manipulation. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023. doi:10.15607/RSS.2023.XIX.047.

[32] H. Chen, Y. Niu, K. Hong, S. Liu, Y. Wang, Y. Li, and K. R. Driggs-Campbell. Predicting object interactions with behavior primitives: An application in stowing tasks. In *7th Annual Conference on Robot Learning*, 2023. URL https://openreview.net/forum?id=VH6WIPF4Sj.

[33] M. Minderer, C. Sun, R. Villegas, F. Cole, K. P. Murphy, and H. Lee. Unsupervised learning of object structure and dynamics from videos. *Advances in Neural Information Processing Systems*, 32, 2019.

[34] N. D. Palo and E. Johns. Dinobot: Robot manipulation via retrieval and alignment with vision foundation models. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[35] W. Liu, A. Daruna, M. Patel, K. Ramachandruni, and S. Chernova. A survey of semantic reasoning frameworks for robotic systems. *Robotics and Autonomous Systems*, 159:104294, 2023.

[36] E. Jang, A. Irpan, M. Khansari, D. Kappler, F. Ebert, C. Lynch, S. Levine, and C. Finn. Bc-z: Zero-shot task generalization with robotic imitation learning. In *Conference on Robot Learning*, pages 991–1002. PMLR, 2022.

[37] N. M. Shafiullah, Z. Cui, A. A. Altanzaya, and L. Pinto. Behavior transformers: Cloning k modes with one stone. *Advances in neural information processing systems*, 35:22955–22968, 2022.

[38] I. Radosavovic, B. Shi, L. Fu, K. Goldberg, T. Darrell, and J. Malik. Robot learning with sensorimotor pre-training. *arXiv preprint arXiv:2306.10007*, 2023.

[39] A. Padalkar, A. Pooley, A. Jain, A. Bewley, A. Herzog, A. Irpan, A. Khazatsky, A. Rai, A. Singh, A. Brohan, et al. Open x-embodiment: Robotic learning datasets and rt-x models. *arXiv preprint arXiv:2310.08864*, 2023.

[40] W. Yan, A. Vangipuram, P. Abbeel, and L. Pinto. Learning predictive representations for deformable objects using contrastive estimation. In *Conference on Robot Learning*, pages 564–574. PMLR, 2021.

[41] Y. Jiang, A. Gupta, Z. Zhang, G. Wang, Y. Dou, Y. Chen, L. Fei-Fei, A. Anandkumar, Y. Zhu, and L. Fan. Vima: General robot manipulation with multimodal prompts. In *Fortieth International Conference on Machine Learning*, 2023.

[42] W. Yuan, C. Paxton, K. Desingh, and D. Fox. Sornet: Spatial object-centric representations for sequential manipulation. In *5th Annual Conference on Robot Learning*, pages 148–157. PMLR, 2021.

[43] X. Chen, K. Zheng, Z. Zeng, C. Kisailus, S. Basu, J. Cooney, J. Pavlasek, and O. C. Jenkins. Manipulation-oriented object perception in clutter through affordance coordinate frames. In *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*, pages 186–193. IEEE, 2022.

[44] C. Ku, C. Winge, R. Diaz, W. Yuan, and K. Desingh. Evaluating robustness of visual representations for object assembly task requiring spatio-geometrical reasoning. *arXiv preprint arXiv:2310.09943*, 2023.

[45] D. Paulius, Y. Huang, R. Milton, W. D. Buchanan, J. Sam, and Y. Sun. Functional object-oriented network for manipulation learning. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2655–2662. IEEE, 2016.

[46] M. Kokic, J. A. Stork, J. A. Haustein, and D. Kragic. Affordance detection for task-specific grasping using deep learning. In *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, pages 91–98. IEEE, 2017.

[47] R. Wu and Y. Zhao. Vat-mart: Learning visual action trajectory proposals for manipulating 3d articulated objects. In *International Conference on Learning Representations (ICLR), 2022*, 2022.

[48] Y. Zhao, R. Wu, Z. Chen, Y. Zhang, Q. Fan, K. Mo, and H. Dong. Dualafford: Learning collaborative visual affordance for dual-gripper manipulation. In *International Conference on Learning Representations*, 2023.

[49] Y. Wang, R. Wu, K. Mo, J. Ke, Q. Fan, L. J. Guibas, and H. Dong. Adaafford: Learning to adapt manipulation affordance for 3d articulated objects via few-shot interactions. In *European conference on computer vision*, pages 90–107. Springer, 2022.

[50] R. Wu, C. Ning, and H. Dong. Learning foresightful dense visual affordance for deformable object manipulation. In *IEEE International Conference on Computer Vision (ICCV)*, 2023.

[51] B. Wen, W. Lian, K. Bekris, and S. Schaal. You only demonstrate once: Category-level manipulation from single visual demonstration. *Robotics: Science and Systems 2022*, 2022.

[52] N. Di Palo and E. Johns. On the effectiveness of retrieval, alignment, and replay in manipulation. *IEEE Robotics and Automation Letters*, 2024.

[53] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4460–4470, 2019.

[54] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

[55] W. Shen, G. Yang, A. Yu, J. Wong, L. P. Kaelbling, and P. Isola. Distilled feature fields enable few-shot language-guided manipulation. *arXiv preprint arXiv:2308.07931*, 2023.

[56] A. Simeonov, Y. Du, A. Tagliasacchi, J. B. Tenenbaum, A. Rodriguez, P. Agrawal, and V. Sitzmann. Neural descriptor fields: Se (3)-equivariant object representations for manipulation. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 6394–6400. IEEE, 2022.

[57] Y.-C. Lin, P. Florence, A. Zeng, J. T. Barron, Y. Du, W.-C. Ma, A. Simeonov, A. R. Garcia, and P. Isola. Mira: Mental imagery for robotic affordances. In *Conference on Robot Learning*, pages 1916–1927. PMLR, 2023.

[58] Y. Ze, G. Yan, Y.-H. Wu, A. Macaluso, Y. Ge, J. Ye, N. Hansen, L. E. Li, and X. Wang. Gnfactor: Multi-task real robot learning with generalizable neural feature fields. In *Conference on Robot Learning*, pages 284–301. PMLR, 2023.

[59] J. Kerr, C. M. Kim, K. Goldberg, A. Kanazawa, and M. Tancik. Lerf: Language embedded radiance fields. In *International Conference on Computer Vision (ICCV)*, 2023.

[60] A. Rashid, S. Sharma, C. M. Kim, J. Kerr, L. Y. Chen, A. Kanazawa, and K. Goldberg. Language embedded radiance fields for zero-shot task-oriented grasping. In *7th Annual Conference on Robot Learning*, 2023. URL https://openreview.net/forum?id=k-Fg8JDQmc.

[61] R.-Z. Qiu, Y. Hu, G. Yang, Y. Song, Y. Fu, J. Ye, J. Mu, R. Yang, N. Atanasov, S. Scherer, and X. Wang. Learning generalizable feature fields for mobile manipulation. *arXiv preprint arXiv:2403.07563*, 2024.

[62] J. Ye, N. Wang, and X. Wang. Featurenerf: Learning generalizable nerfs by distilling foundation models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8962–8973, 2023.

[63] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *arXiv preprint arxiv:2006.11239*, 2020.

[64] F. Xiang, Y. Qin, K. Mo, Y. Xia, H. Zhu, F. Liu, M. Liu, H. Jiang, Y. Yuan, H. Wang, L. Yi, A. X. Chang, L. J. Guibas, and H. Su. SAPIEN: A simulated part-based interactive environment. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[65] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Proceedings of Robotics: Science and Systems (RSS)*, 2023.

GenDP: 3D Semantic Fields for Category-Level Generalizable Diffusion Policy

Yixuan Wang[1], **Guang Yin**[2], **Binghao Huang**[1], **Tarik Kelestemur**[3], **Jiuguang Wang**[3], **Yunzhu Li**[1]

[1]Columbia University, [2]University of Illinois, Urbana-Champaign, [3]Boston Dynamics AI Institute

Contents

1 Method **2**

1.1 Detailed Comparisons with Prior Semantic Fields 2

1.2 Detailed Comparisons with Prior Diffusion Policies 2

1.3 Detailed Explanation on Geometric Ambiguity 3

1.4 Detailed Explanation on Geometric Details 3

1.5 PointNet++ Details 3

1.6 Reference Descriptors 3

2 Experiments Details **4**

2.1 Setup 4

2.2 Demonstrations 5

2.3 Training 5

2.4 Inference 5

2.5 Evaluation 5

3 Additional Experiments **7**

3.1 Quantitative Experiment Table 7

3.2 Extension to ACT 7

3.3 Ablation on Feature Extraction Backbone 8

3.4 More Baselines 8

3.4.1 SORNet 8

3.4.2 DINOBot 9

3.4.3 Results 9

3.5 t-SNE 10

3.6 Comparisons with Different RGBD Encoder 10

3.7 Comparisons with Semantic-Informed Encoder 11

3.8 Ablation on Different Views 11

8th Conference on Robot Learning (CoRL 2024), Munich, Germany.

1 Method

1.1 Detailed Comparisons with Prior Semantic Fields

Compared to prior works on semantic fields and representation, our paper offers following benefits:

- **Generalizable Representation**: Some existing semantic fields or representations are trained on small-scale datasets and are challenging to transfer to novel scenes or object categories. For example, Dense Object Nets (DON) [1] are trained via contrastive learning, which requires foreground and background separation and a careful selection of positive and negative pairs. This means that deploying a DON in a new environment or on new object categories requires additional training efforts. Similarly, DSR-Net [2] trains its representation on datasets collected by heuristic policy, requiring manual designs for different scenes. Therefore, deploying DSR-Net on new objects and scenarios also requires extra training efforts. In contrast, our pipeline does not need additional training and can be applied to a diverse set of object categories and scenes by leveraging visual foundational models.
- **Efficiency**: Several prior works build semantic fields using visual foundational models. However, some are not efficient enough for imitation learning, which needs to obtain visual feedback in real time. For example, F3RM [3] takes about 90 seconds for each scene, and Feature 3DGS [4] training time is around 200 seconds, according to its Supplementary Figure A. In contrast, our 3D semantic fields only need 0.166 seconds per frame, and the total inference time for each step, including the diffusion process, is 0.226 seconds.
- **Sparse Views**: Existing semantic fields need an extensive number of camera views to build quality fields, which limits their usage in imitation learning. For example, F3RM [3] requires 50 images to scan the workspace, and Feature 3DGS [4] needs 80 views, as specified in its Supplementary Section D. ConceptGraphs [5] uses the Replica dataset, which contains 2,000 views for each scene. In contrast, we use only four cameras placed around the workspace to build our 3D semantic fields, enabling us to obtain real-time visual feedback and achieve a control frequency of 4-5 Hz.

1.2 Detailed Comparisons with Prior Diffusion Policies

Compared to prior works on diffusion policies, our paper offers the following benefits:

- **Category-Level Generalization**: Although existing diffusion policies based on 3D point clouds, such as 3D Diffuser Actor [6], demonstrate certain levels of generalization capabilities, they do not yet exhibit robust enough category-level generalization capabilities. In our work, we introduce 3D semantic fields that can significantly boost performance and facilitate category-level generalization, as shown in Figure 4 (iv) of our paper.
- **Semantic Understanding**: Prior works, such as 3DP [7], have explored using raw 3D point clouds as inputs for diffusion policies. However, we found that geometric information can be ambiguous and miss important details, leading to failure in many daily tasks. In contrast, we provide additional semantic features to guide the policy to focus on semantically meaningful object parts to accomplish various tasks.
- **Flexible Action Representation**: We also note that 3D Diffuser Actor predicts end-effector keyposes instead of a continuous trajectory, requiring manual keypose definition for each task, as shown in its Appendix A.6. Additionally, keypose representation only specifies the starting and end poses, which may not suffice for tasks requiring the control over the whole trajectory, such as scooping coffee beans out and transporting them (our Use Spoon task). In our work, we predict actions over the whole trajectory, which is more flexible for different tasks.

1.3 Detailed Explanation on Geometric Ambiguity

In our paper, geometric ambiguity means that geometric information alone cannot distinguish between different underlying object states. For example, in the Open Pen task, the raw point cloud data of the pen is ambiguous because rotating the pen 180 degrees can result in a very similar point cloud. This makes it difficult to determine the position of the cap—a critical piece of state-related information for the success of the downstream cap-opening task.

One might consider incorporating RGB information into the point cloud. However, RGB information is sensitive to appearance variations, limiting its ability for category-level generalization.

Therefore, in our work, it is the semantic information that helps the policy to disambiguate geometric ambiguity and learn the correct correlation between observations and robotic actions. We combine geometric information with the semantic field—two necessary components for category-level generalization in complex manipulation tasks. Geometric information alone is ambiguous about the underlying states, while the semantic field helps identify where the caps are on different pens with varying appearances.

1.4 Detailed Explanation on Geometric Details

Compared to RGB data, 3D geometric data is robust to environmental and appearance changes, helping the diffusion policy better generalize to unseen instances, as shown in Figure 4 and Supplementary Figure 1. However, geometric information may not capture many important details, whereas semantic fields can highlight the right details and help the policy learn the correct correlation between observations and robot actions.

For example, in the Place Can task, even if there are slight geometric variations in the can tab, it can be difficult for the policy to attend to that location. Being aware of geometric details like the can tab, the trained policy can place the can in the correct pose instead of an upside-down position. In our experiments, we show that semantic fields can significantly boost the task success rate in the Place Can and Use Toothbrush tasks.

1.5 PointNet++ Details

We use PointNet++ to encode one latent vector from the point clouds [8]. There are several key details regarding its usage.

- No BatchNorm Layer: In the diffusion policy implementation, exponential moving averages (EMA) of the model parameters are used during the testing, which is not compatible with BatchNorm.
- Point Set Abstraction: Point set abstraction is an important operation in PointNet++. In our network, we deploy three layers of point set abstraction. For the first set abstraction, we have 512 groups with a radius of 0.04 and 32 points. For the second set abstraction, we have 128 groups with a radius of 0.08 and 64 points. For the last set abstraction, all sampled points are grouped to extract one latent vector.

1.6 Reference Descriptors

For one object category, we use four images to obtain reference descriptors, as shown in Figure 1. We start by acquiring four images containing the target object category using the Internet or any arbitrary phone camera. We then obtain a single descriptor by selecting one keypoint on each of the four images and averaging their corresponding features. Therefore, we only need to select $N \times 4$ keypoints for each object category, where N is the number of reference descriptors. These reference descriptors can be used for the entire task (e.g., 100 training demonstrations and testing scenes), making the labeling effort light.

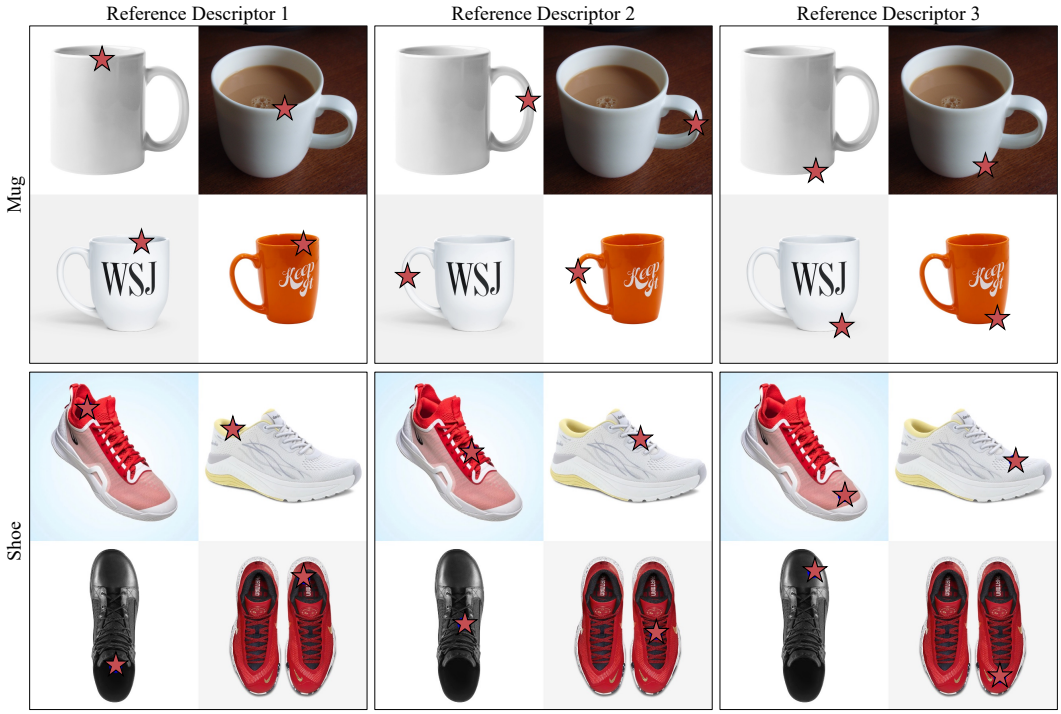


Figure 1: **Reference Images and Descriptors Example.** For one object category, we will obtain N reference descriptors, with $N = 3$ in these examples. For each descriptor, we will select one keypoint representing a specific object part from each figure. The keypoint is denoted as a red pentagram in this figure. We then average the four descriptors to obtain one reference descriptor.

2 Experiments Details

2.1 Setup

We summarize object categories and the number of object instances we considered in Table 1.

Task Category	Task Name	#Demo	Object	#Train Instances	#Test Instances
Sim.	Hang Mug	200	Mug	4	2
	Insert Pencil	100	Pencil	2	1
Geo. Details	Place Can	60	Soda Can	3	3
	Use Toothbrush	60	Toothbrush	1	4
Cat. Gen.	Align Shoes	60	Shoe	3	5
	Use Spoon	60	Spoon	2	2
Geo. Ambi.	Collect Knife	60	Knife	1	2
	Open Pen	60	Marker Pen	1	3

Table 1: **Task Details Summary.** This table summarizes our task categories, specific tasks, objects, the number of demonstrations, training instances, and testing instances. We evaluate our framework on eight tasks and eight object categories, where each object category includes several instances with diverse appearances and shapes. We divide our tasks into four categories to study our method's performance under various task types and scenarios, such as simulation and the real world.

Here is a list of task descriptions:

- **Hang Mug (Simulation)**: Hang a randomly placed mug on a fixed mug tree.
- **Pencil Insertion (Simulation)**: Pick up a pencil from the table and insert it into the pencil sharpener.

- **Collect Knife**: Collect a lying knife into an open container, and the knife direction is randomly chosen.
- **Open Pen**: Bimanually grasp the pen and open it.
- **Flip Can**: Flip a lying soda can and place it upright.
- **Use Toothbrush**: Grasp the toothbrush and spread the toothpaste on it.
- **Align Shoes**: Push shoes towards left.
- **Use Spoon**: Grasp spoon and scoop materials.

Due to space limitations, in all tables, we will use the following abbreviation:

- **Sim.**: Simulation
- **Geo. Details**: Geometry Details
- **Geo. Ambi.**: Geometry Ambiguity
- **Cat. Gen.**: Category Generalization

2.2 Demonstrations

In the simulation, we use SAPIEN to create simulation environments. We build an oracle policy to generate demonstrations in the simulation. The generated dataset contains robot states, multi-view RGBD observations, and robot actions.

In the real world, the data collection pipeline is similar, with additional calibration steps. We define the ChArUco board as the world coordinate and calibrate the camera poses to this coordinate. Then, we manually calibrate the transformation between the robot and the world coordinate. After the calibration phase, the data collection pipeline records multi-view RGBD observations, robot states, and robot actions.

2.3 Training

Before training, we preprocess multi-view RGBD observations into 3D semantic fields using the frozen DINOv2 encoder and save them to CPU memory. During training, the 3D semantic fields and corresponding ground truth actions are randomly sampled and fed into the diffusion policy. The training of the diffusion policy follows the practice outlined in the original Diffusion Policy paper [9].

2.4 Inference

During inference, the diffusion policy takes in one frame of multi-view RGBD observation and converts it into 3D semantic fields. Then, the 3D semantic fields are input into the diffusion policy to predict an action sequence.

We measure our inference time on the knife-collecting task, repeating the inference steps 10 times and computing the average time. Each inference step takes 0.226 seconds. The predicted action sequence is at 10 Hz, meaning the time difference between two consecutive actions in the action sequence is 0.1 seconds. This action sequence is then fed into the low-level controller, which operates at 50 Hz.

2.5 Evaluation

Here we list the evaluation criteria for each task:

- Hang Mug (Sim): If the final height of the mug is larger than 0.1m and lower than 0.3m, and the xy-space distance between the mug holder and the mug is lower than 0.1m, the reward is 1. Otherwise, the reward is 0.

- Pencil Insertion (Sim): If the pencil's final height is larger than 0.05m, and the xy-space angle between the pencil and the hole is smaller than $\arccos(0.8)$, the reward is 1. Otherwise, the reward is 0.

- Place Can: The policy rollout is only successful if the can is grasped and placed upright on the table.

- Use Toothbrush: The policy rollout is only successful if the toothbrush is grasped and the head of the toothbrush is aligned with the toothpaste tip.

- Align Shoes: The policy rollout is only successful if the shoe is rotated towards the left.

- Use Spoon: The policy rollout is only successful if the robot grasps the spoon, scoops the coffee beans, and pours into the bowl.

- Collect Knife: The policy rollout is only successful if the robot grasps the knife handle and puts it in the box.

- Open Pen: The policy rollout is only successful if the robot first grasps the pen body using one hand, and then pulls out the pen tap using another hand.

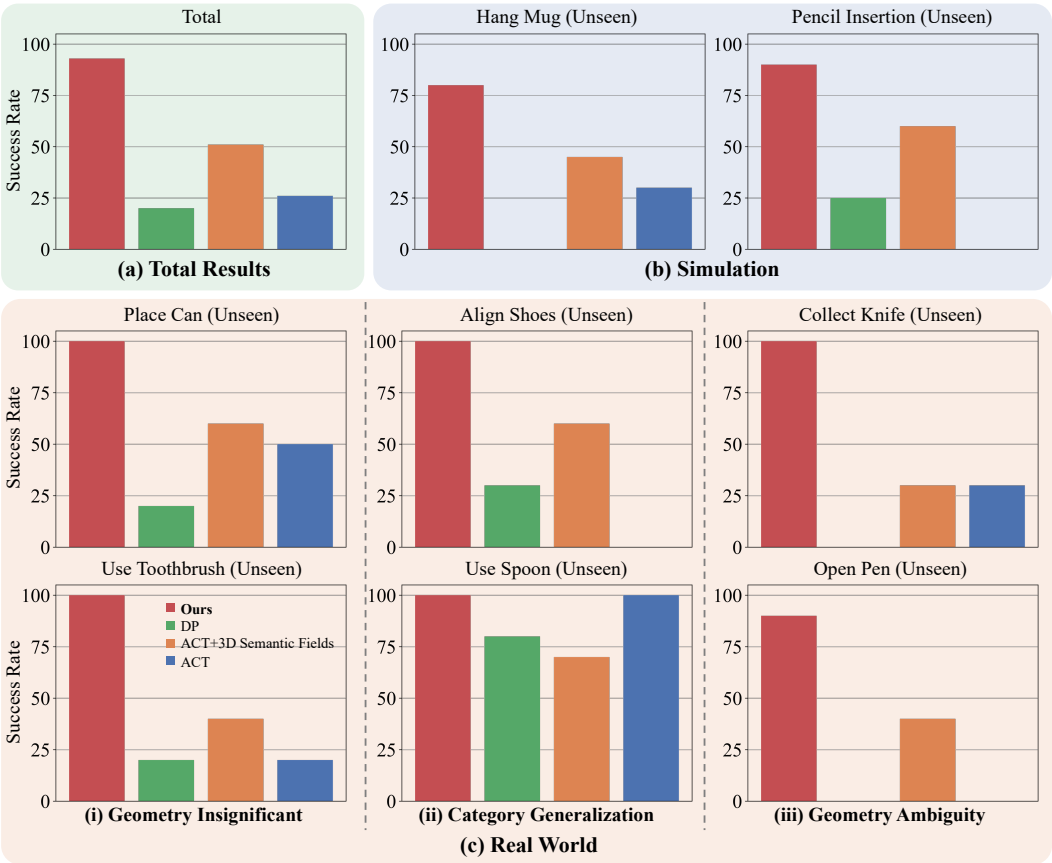


Figure 2: **Success Rate of Extension to ACT.** We measure the success rate of our method, diffusion policy, ACT with our 3D semantic fields, and ACT on unseen instances across different tasks. We found that our 3D semantic fields can help other imitation learning methods like ACT to generalize to novel instances as well. However, in practice, we choose the diffusion policy with 3D semantic fields since it performs the best across different tasks.

3 Additional Experiments

3.1 Quantitative Experiment Table

As mentioned in Section 4.2 of the main paper, we present a detailed table for Figure 4. This table provides more concrete results regarding our comparisons with the baseline methods. We can see that, although the diffusion policy performs well for the seen instances, it fails to generalize to novel instances. Compared with our method without semantics, our method also demonstrates the effectiveness of incorporating semantic information.

Task Category	Simulation			
Task Name	Hang Mug		Insert Pencil	
Instances	Seen	Unseen	Seen	Unseen
Ours	**95% (19/20)**	80% (16/20)	**95% (19/20)**	**90% (18/20)**
Ours w/o Semantics	65% (13/20)	**85% (17/20)**	40% (8/20)	40% (8/20)
Diffusion Policy	**95% (19/20)**	0% (0/20)	**95% (19/20)**	25% (5/20)
Diffusion Policy w/ RGBD	0% (0/20)	0% (0/20)	15% (3/20)	20% (4/20)

Task Category	Geometry Ambiguity		Geometry Details	
Task Name	Collect Knife	Open Pen	Place Can	Use Toothbrush
Instances	Unseen	Unseen	Unseen	Unseen
Ours	**100% (9/10)**	**90% (9/10)**	**100% (10/10)**	**100% (10/10)**
Ours w/o Semantics	50% (5/10)	50% (5/10)	60% (6/10)	30% (3/10)
Diffusion Policy	0% (0/10)	0% (0/10)	20% (2/10)	20% (2/10)
Diffusion Policy w/ RGBD	60% (6/10)	20% (2/10)	0% (0/10)	20% (2/10)

Task Category	Category Generalization			
Task Name	Collect Knife	Open Pen	Total (Seen)	Total (Unseen)
Instances	Unseen	Unseen		
Ours	**100% (10/10)**	**100% (10/10)**	**95% (38/40)**	**93% (93/100)**
Ours w/o Semantics	60% (6/10)	90% (9/10)	53% (21/40)	59% (59/100)
Diffusion Policy	30% (3/10)	80% (8/10)	**95% (38/40)**	20% (20/100)
Diffusion Policy w/ RGBD	10% (1/10)	70% (7/10)	8% (3/40)	22% (22/100)

Table 2: **Success Rate.** The method was evaluated across eight tasks. We observe that the diffusion policy performs similarly to our method in the seen environments but shows markedly worse performance in unseen instances. In addition, compared with ours without semantics, our method shows a better performance across different tasks and demonstrates the necessity of semantic information. These results underscore our policy's capability to achieve category-level generalization and encode semantic information.

3.2 Extension to ACT

We also extend our representation to other imitation learning methods, such as ACT [10]. Figure 2 shows the experiment results. First, we found that our representation helps other imitation learning algorithms generalize to novel instances. This is because raw RGB images are sensitive to environmental factors and object appearances, while our 3D semantic fields are robust against environmental changes and novel instances by explicitly using geometric and semantic information. Second, we found that our method outperforms ACT. We believe the main reason is that the diffusion policy is better at capturing multi-modal demonstration distribution compared to ACT, which is quite common in the real world. In summary, our representation can be applied to different imitation learning methods, but we choose the diffusion policy because it performs the best in practice.

Figure 3: **3D Semantic Fields Visualization.** We visualize 3D semantic fields in the scene, similar to Figure 6 in the main paper. For ours with DINOv2, object parts from different instances are highlighted consistently. For example, shoe heads, book titles, and mug handles are highlighted in these scenes. In contrast, ours with CLIP and ours with ViT fail to highlight object parts consistently. Instead, their heatmap distributes over the scene and has no distinctions among object parts.

3.3 Ablation on Feature Extraction Backbone

Originally, we used DINOv2 as the feature backbone [11]. To study the influence of different feature backbones on performance, we replace DINOv2 with other feature backbones. Other than DINOv2, CLIP and Vision Transformer (ViT) are two of the most common image feature extractors [12, 13]. Therefore, we replace DINOv2 with CLIP and ViT and compare their performance. First, we try to qualitatively understand what the 3D semantic fields will look like with different feature backbones. Second, we quantitatively show the performance of the whole diffusion policy pipeline when the feature backbones are different.

Similar to Figure 6 in the main paper, we use the same 3D semantic field construction and visualization pipeline with different feature backbones. We visualize the 3D semantic fields of different object parts in Figure 3. We found that ours with DINOv2 can consistently highlight the object parts, such as shoe heads, book titles, and mug handles, while ours with CLIP and ours with ViT fail to do so. Without the capability to distinguish semantically meaningful parts of the object, ours with ViT and ours with CLIP fail to generalize to unseen instances, distinguish geometric ambiguities, and attend to subtle geometric details. This qualitative result justifies our usage of DINOv2 to generate useful 3D semantic fields.

Table 3 shows our quantitative results, where the experiment process is the same as the main paper. The table shows our method's average success rate is 94%, which is the best among all methods. As this table shows, ours with CLIP's average success rate is 57%, and ours with ViT's average success rate is 55%. This result is close to ours without semantics (58%) as shown in Table 1. This indicates that CLIP and ViT do not encode useful semantic information like object parts. In contrast, our method can highlight parts of the object that are critical for the task's success.

3.4 More Baselines

We compare with SORNet and DINOBot [14, 15]. We will describe how we compare our work with these two baselines in our setting respectively in Section 3.4.1 and Section 3.4.2. Then we present results and analyze them in Section 3.4.3.

3.4.1 SORNet

SORNet introduces an object-centric representation that can be generalized to objects unseen during training. Specifically, SORNet can take in observation from more than one view, where depth is

Task Category	Task Name	Instances	Ours	Ours with CLIP	Ours with ViT
Sim.	Hang Mug	Seen	**95% (19/20)**	55% (11/20)	80% (16/20)
		Unseen	**80% (16/20)**	50% (10/20)	70% (14/20)
	Insert Pencil	Seen	**95% (19/20)**	70% (14/20)	55% (11/20)
		Unseen	**90% (18/20)**	45% (9/20)	30% (6/20)
Geo. Ambi.	Collect Knife	Unseen	**100% (10/10)**	30% (3/10)	30% (3/10)
	Open Pen	Unseen	**100% (10/10)**	50% (5/10)	40% (4/10)
Geo. Details	Place Can	Unseen	**100% (10/10)**	40% (4/10)	50% (5/10)
	Use Toothbrush	Unseen	**90% (9/10)**	50% (5/10)	50% (5/10)
Cat. Gen.	Align Shoes	Unseen	**100% (10/10)**	90% (9/10)	40% (4/10)
	Use Spoon	Unseen	**100% (10/10)**	100% (10/10)	90% (9/10)
Total (Unseen)			**94% (131/140)**	57% (80/140)	55% (77/140)

Table 3: **Success Rate for Different Feature Backbones.** The method was evaluated across eight tasks and compared against different feature backbones. We use the average success rate as our evaluation metric. Compared with ours with CLIP and ours with ViT, our method consistently outperforms them. This demonstrates that DINOv2 could encode semantic information, such as parts of object that are important for task completion.

optional. It also takes in canonical object views and outputs corresponding embedding for each query object. For fair comparisons, we input RGBD observations from four viewpoints and one canonical object view into SORNet and obtain one embedding. Then we input this embedding to diffusion policy, the same as what did for our approach. We do not use pre-trained models provided by SORNet, since the observation modality and training domain are very different. Instead, we train the whole model from scratch.

3.4.2 DINOBot

DINOBot introduces a single-shot imitation learning framework that leverages visual features from foundational vision models. They first record the key end-effector pose and corresponding RGBD observations. Then they record a motion sequence. During testing, they will align current RGBD observations with recorded key RGBD observations to make sure the end-effector has arrived at the key pose. Then they will replay the motion sequence.

In our implementation, we choose one of our perspective views as the observation view. For the rest of the implementation, we follow the DINOBot.

3.4.3 Results

As shown in Table 4, SORNet performs badly for both seen and unseen instances. As mentioned by SORNet's authors, SORNet needs to be trained on a large-scale dataset with diverse object appearance and scene layouts. Specifically, they trained SORNet on two datasets, which contain 405 training objects and 330 training objects respectively. However, we only assume access to a demonstration dataset collected over one or a handful of objects. SORNet cannot effectively extract useful representation in this low-data regime, while our representation could generalize to unseen instances given the same amount of data.

As shown in Table 4, DINOBot cannot accomplish the tasks and fails in both seen and unseen scenarios. The reasons are twofold. First, DINOBot merely replays the recorded trajectory after the 6-DoF alignment, which limits its applicability in tasks that require adaptive adjustments based on the positions and shapes of objects. Second, DINOBot has only tackled single-arm manipulation tasks using a wrist-mounted camera, which restricts its extension to more complicated bimanual tasks (e.g., pen opening, as we demonstrated using our method). In contrast, our method represents

Task Category	Task Name	Instances	Ours	SORNet [14]	DINOBot [15]
Sim.	Hang Mug	Seen	**95% (19/20)**	0% (0/20)	0% (0/20)
		Unseen	**80% (16/20)**	0% (0/20)	0% (0/20)
	Insert Pencil	Seen	**95% (19/20)**	0% (0/20)	0% (0/20)
		Unseen	**90% (18/20)**	0% (0/20)	0% (0/20)
Geo. Ambi.	Collect Knife	Unseen	**100% (10/10)**	10% (1/10)	0% (0/10)
	Open Pen	Unseen	**100% (10/10)**	0% (0/10)	N/A% (N/A)
Geo. Details	Place Can	Unseen	**100% (10/10)**	20% (2/10)	0% (0/10)
	Use Toothbrush	Unseen	**90% (9/10)**	0% (0/10)	0% (0/10)
Cat. Gen.	Align Shoes	Unseen	**100% (10/10)**	0% (0/10)	0% (0/10)
	Use Spoon	Unseen	**100% (10/10)**	40% (4/10)	0% (0/10)
Total (Unseen)			**94% (131/140)**	5% (7/140)	0% (0/130)

Table 4: **Success Rate for Additional Representation and Imitation Learning Baselines.** The method was evaluated across eight tasks and compared with SORNet and DINOBot. We use the average success rate as our evaluation metric. The low success rate of SORNet implies that it needs large-scale data for pre-training, while we only assume access to demonstrations recorded in several hours. DINOBot's low success rate implies it is not as flexible as our method for action representation and task specification. There is no result for the opening pen task because of DINOBot's assumption of single-arm manipulation.

the action as a sequence of end-effector poses, which can be easily extended to a dual-arm setup and accomplish a more diversified set of tasks.

3.5 t-SNE

In addition, we also analyze whether our semantic fields could carry desired semantic information, as shown in Figure 4. We initially sample 3D grid points in the workspace and select the top 100 points with the highest similarity score for each semantic field. For the example shown in Figure 4, there are 500 points in total. Then, we query the descriptor fields and obtain semantic features corresponding to the selected points. We reduce the high-dimensional semantic features to a two-dimensional plane using t-distributed stochastic neighbor embedding (t-SNE). We observe that all points are distinctly separated, and each blob corresponds to one object part. This visualization shows that our semantic fields have a clear semantic meaning for each channel, which helps the policy to achieve category-level generalization.

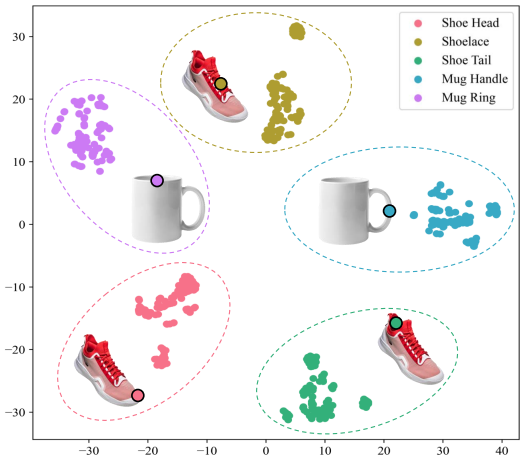


Figure 4: **t-SNE for Semantic Features with High Similarity.** Points in similarity fields are selected based on the top k similarity scores. We compute their corresponding semantic features and project them into two-dimensional space using t-SNE. It is clearly observed that all features are clustered and separated from each other. Each cluster corresponds to a feature that has semantic meaning, such as mug rings and mug handles.

3.6 Comparisons with Different RGBD Encoder

We also tried different RGBD encoders other than the standard encoder used for RGB. Specifically, we tested RD3D [16] as a different RGBD encoder (DP+RD3D) and evaluated it on two simulation tasks using the same dataset and task setup as the main paper. Our results are summarized in Table 5. We found that DP+RD3D fails in both tasks. We hypothesize that this is because RD3D was designed for object detection, which may not be particularly suitable for diffusion policy.

Method	Hang Mug (Unseen)	Insert Pencil (Unseen)	Total (Unseen)
DP	0% (0/20)	25% (5/20)	12.5% (5/40)
DP+RGBD	0% (0/20)	20% (4/20)	10% (4/40)
DP+RD3D	0% (0/20)	0% (0/20)	0% (0/40)
Ours w/o Semantics	**85% (17/20)**	40% (8/20)	62.5% (25/40)
Ours	80% (16/20)	**90% (18/20)**	**85% (34/40)**

Table 5: **Success Rate for DP+RD3D [16].** DP+RD3D was evaluated across two simulation tasks with the same setting as the main paper. We use the average success rate as our evaluation metric. We found that DP+RD3D fails in both tasks. We hypothesize that this is because RD3D was designed for object detection, which may not be particularly suitable for diffusion policy.

3.7 Comparisons with Semantic-Informed Encoder

Similar to our approach, we first extract DINOv2 feature maps from RGB and then build the semantic map by comparing to reference features, the same as in our paper. We then append the semantic map to the RGB and input it into the diffusion policy (DP+Semantic). Our results are summarized in Table 6

	Hang Mug (Unseen)	Insert Pencil (Unseen)	Total (Unseen)
DP	0% (0/20)	25% (5/20)	12.5% (5/40)
DP+RGBD	0% (0/20)	20% (4/20)	10% (4/40)
DP+Semantic	0% (0/20)	75% (15/20)	37.5% (15/40)
Ours w/o Semantics	**85% (17/20)**	40% (8/20)	62.5% (25/40)
Ours	80% (16/20)	**90% (18/20)**	**85% (34/40)**

Table 6: **Success Rate for DP+Semantic.** DP+Semantic was evaluated across two simulation tasks with the same setting as the main paper. We use the average success rate as our evaluation metric. We found that DP+Semantic performs better than DP, demonstrating that additional semantic information can guide the diffusion policy to focus on object parts important for task success. But it still performs worse than our approach.

We found that DP+Semantic performs better than DP, demonstrating that additional semantic information can guide the diffusion policy to focus on object parts important for task success. However, DP+Semantic still learns spurious correlations between RGB data and robotic actions, making it sensitive to appearance variations when encountering unseen instances from the same category. In contrast, our method utilizes 3D semantic fields that are robust to such variations and enable category-level generalization of the diffusion policy.

3.8 Ablation on Different Views

We provide additional experiments of training policy with different numbers of views, and our results are summarized in Table 7. We found that the performance of our method is not significantly affected by the number of camera views.

4 Detailed Comparisons with Prior Works

- *Functional Object-Oriented Network for Manipulation Learning* [17]: This work proposes a structured knowledge representation and generates motion sequences based on the representation. While this work represents the motion sequence as motion harmonics, our work is more flexible in action and task representation. In addition, this work does not show extensive real-world deployment results, while we focus more on the real-world deployment.
- *Affordance Detection for Task-Specific Grasping Using Deep Learning* [18]: This work uses convolutional neural networks to detect object affordance, class, and orientation for

Methods	Hang Mug (Unseen)	Pencil Insertion (Unseen)
1 View	85% (17/20)	90% (18/20)
2 Views	90% (18/20)	95% (19/20)
3 Views	75% (15/20)	85% (17/20)
4 Views	80% (16/20)	90% (18/20)
Total	82.5% (66/80)	90% (72/80)

Table 7: **Success Rate for Different Number of Views.** This table illustrates the success rates of our method with different numbers of views evaluated on two unseen tasks: Hang Mug and Pencil Insertion. We found that the performance of our method is not significantly affected by the number of camera views.

task-specific tasks. Although it shows generalization capabilities, it is still restricted to grasping tasks. In contrast, our framework supports a wider range of tasks.

- *SORNet: Spatial Object-Centric Representations for Sequential Manipulation* [14]: SORNet proposes an object-centric representation from RGB observations. We note three major differences from SORNet to our work. First, SORNet is object-centric, while ours is scene-representation, which needs less prior information. Second, SORNet builds representation from RGB, which makes it hard to generalize to novel domains, environments, and backgrounds, but our 3D semantic fields show category-level generalization capabilities. Lastly, through experiments, we found that SORNet might be better if pre-trained on a large dataset. In our work, we only assume access to a demonstration dataset of one or a handful of objects, while SORNet needs a larger dataset for pre-training to have a good performance.
- *Manipulation-Oriented Object Perception in Clutter Through Affordance Coordinate Frames* [19]: This work introduces the Affordance Coordinate Frame (ACF) for category-level pose estimation. It shows impressive category-level generalization capabilities. However, this work uses pre-defined motion primitives (grasp, pour, stir) for manipulation, which limits applicable tasks. In contrast, we do not assume any prior motion primitives, which is more flexible for action representation.
- *StructDiffusion: Object-Centric Diffusion for Semantic Rearrangement of Novel Objects* [20]: This work uses a diffusion model and object-centric transformer to construct goal structures given current point clouds and language inputs. They mainly focus on picking and placing tasks using modular methods, including the grasping planner and motion planner. Their modular method might not transfer to other tasks and other types of objects like deformable objects and granular objects easily. Without assumptions for target tasks and objects, our framework is flexible for object types and task specifications.
- *A Survey of Semantic Reasoning Frameworks for Robotic Systems* [21]: We position our work as instance-level semantic reasoning according to this survey. In our work, we propose 3D semantic fields, which can differentiate the object parts using foundational vision models.
- *Learning Generalizable Feature Fields for Mobile Manipulation* [22]: GeFF is a concurrent work using Generelizable NeRF for mobile manipulation [23]. There are two major differences between our work and GeFF. First, we do not distill feature fields from foundational vision models to avoid the loss in generalization capabilities. Second, GeFF uses open-push-close action sequences to accomplish tasks, while our policy predicts the end-effector trajectory, which is more flexible.
- *Evaluating Robustness of Visual Representations for Object Assembly Task Requiring Spatio-Geometrical Reasoning* [24]: This work mainly focuses on object assembly tasks by evaluating the robustness of visual representations from RGB observations. First, since RGB observation is sensitive to environment and instance variance, our method proposes

to use 3D semantic fields to help the generalization of diffusion policy. Second, instead of focusing on assembly tasks, our framework is tested on a broader set of tasks.

- *You Only Demonstrate Once: Category-Level Manipulation from Single Visual Demonstration* [25]: You Only Demonstrate Once introduces the category-level generalizable last-inch policy using model-free pose estimation and trajectory transformation. There are two major differences. First, their behavior cloning algorithm is replaying the trajectory using the object's pose, which limits their applicability to tasks requiring adaptive adjustments based on the positions and shapes of objects. In contrast, we represent the trajectory as a sequence of end-effector poses, which is more flexible for task specification. In addition, they assume the object is rigidly attached to the end-effector, which is not applicable to some tasks like pushing the shoe. In summary, many of our tasks are not doable using this framework.
- *On the Effectiveness of Retrieval, Alignment, and Replay in Manipulation* and *DINOBot: Robot Manipulation via Retrieval and Alignment with Vision Foundation Models* [15, 26]: This line of work from the set of authors proposes a generalizable imitation learning framework, following phases of retrieval, alignment, and replay. DINOBot is the latest work enhancing the prior work using vision foundation models. First, DINOBot merely replays the recorded trajectory after the 6-DoF alignment, which limits its applicability in tasks that require adaptive adjustments based on the positions and shapes of objects. Second, DINOBot has only tackled single-arm manipulation tasks using a wrist-mounted camera, which restricts its extension to more complicated bimanual tasks (e.g., pen opening, as we demonstrated using our method). In contrast, our method represents the action as a sequence of end-effector poses, which can be easily extended to a dual-arm setup and accomplish a more diversified set of tasks.

References

[1] P. R. Florence, L. Manuelli, and R. Tedrake. Dense object nets: Learning dense visual object descriptors by and for robotic manipulation. *arXiv preprint arXiv:1806.08756*, 2018.

[2] Z. Xu, Z. He, J. Wu, and S. Song. Learning 3d dynamic scene representations for robot manipulation. *arXiv preprint arXiv:2011.01968*, 2020.

[3] W. Shen, G. Yang, A. Yu, J. Wong, L. P. Kaelbling, and P. Isola. Distilled feature fields enable few-shot language-guided manipulation. *arXiv preprint arXiv:2308.07931*, 2023.

[4] S. Zhou, H. Chang, S. Jiang, Z. Fan, Z. Zhu, D. Xu, P. Chari, S. You, Z. Wang, and A. Kadambi. Feature 3dgs: Supercharging 3d gaussian splatting to enable distilled feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21676–21685, 2024.

[5] Q. Gu, A. Kuwajerwala, S. Morin, K. M. Jatavallabhula, B. Sen, A. Agarwal, C. Rivera, W. Paul, K. Ellis, R. Chellappa, et al. Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5021–5028. IEEE, 2024.

[6] T.-W. Ke, N. Gkanatsios, and K. Fragkiadaki. 3d diffuser actor: Policy diffusion with 3d scene representations. *Arxiv*, 2024.

[7] Y. Ze, G. Zhang, K. Zhang, C. Hu, M. Wang, and H. Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. *arXiv preprint arXiv:2403.03954*, 2024.

[8] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv preprint arXiv:1706.02413*, 2017.

[9] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Proceedings of Robotics: Science and Systems (RSS)*, 2023.

[10] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Proceedings of Robotics: Science and Systems (RSS)*, 2023.

[11] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, R. Howes, P.-Y. Huang, H. Xu, V. Sharma, S.-W. Li, W. Galuba, M. Rabbat, M. Assran, N. Ballas, G. Synnaeve, I. Misra, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.

[12] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision. In M. Meila and T. Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

[13] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

[14] W. Yuan, C. Paxton, K. Desingh, and D. Fox. Sornet: Spatial object-centric representations for sequential manipulation. In *5th Annual Conference on Robot Learning*, pages 148–157. PMLR, 2021.

[15] N. D. Palo and E. Johns. Dinobot: Robot manipulation via retrieval and alignment with vision foundation models. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[16] Q. Chen, Z. Zhang, Y. Lu, K. Fu, and Q. Zhao. 3-d convolutional neural networks for rgb-d salient object detection and beyond. *IEEE Transactions on Neural Networks and Learning Systems*, 35(3):4309–4323, 2022.

[17] D. Paulius, Y. Huang, R. Milton, W. D. Buchanan, J. Sam, and Y. Sun. Functional object-oriented network for manipulation learning. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2655–2662. IEEE, 2016.

[18] M. Kokic, J. A. Stork, J. A. Haustein, and D. Kragic. Affordance detection for task-specific grasping using deep learning. In *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, pages 91–98. IEEE, 2017.

[19] X. Chen, K. Zheng, Z. Zeng, C. Kisailus, S. Basu, J. Cooney, J. Pavlasek, and O. C. Jenkins. Manipulation-oriented object perception in clutter through affordance coordinate frames. In *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*, pages 186–193. IEEE, 2022.

[20] W. Liu, Y. Du, T. Hermans, S. Chernova, and C. Paxton. Structdiffusion: Language-guided creation of physically-valid structures using unseen objects. In *RSS 2023*, 2023.

[21] W. Liu, A. Daruna, M. Patel, K. Ramachandruni, and S. Chernova. A survey of semantic reasoning frameworks for robotic systems. *Robotics and Autonomous Systems*, 159:104294, 2023.

[22] R.-Z. Qiu, Y. Hu, G. Yang, Y. Song, Y. Fu, J. Ye, J. Mu, R. Yang, N. Atanasov, S. Scherer, and X. Wang. Learning generalizable feature fields for mobile manipulation. *arXiv preprint arXiv:2403.07563*, 2024.

[23] J. Ye, N. Wang, and X. Wang. Featurenerf: Learning generalizable nerfs by distilling foundation models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8962–8973, 2023.

[24] C. Ku, C. Winge, R. Diaz, W. Yuan, and K. Desingh. Evaluating robustness of visual representations for object assembly task requiring spatio-geometrical reasoning. *arXiv preprint arXiv:2310.09943*, 2023.

[25] B. Wen, W. Lian, K. Bekris, and S. Schaal. You only demonstrate once: Category-level manipulation from single visual demonstration. *Robotics: Science and Systems 2022*, 2022.

[26] N. Di Palo and E. Johns. On the effectiveness of retrieval, alignment, and replay in manipulation. *IEEE Robotics and Automation Letters*, 2024.