Affordance-Centric Policy Decomposition π :
Generalisable and Sample Efficient Robot Policy Learning for Multi-Object, Long-Horizon Manipulation

Abstract:

Long-horizon manipulation tasks involving multiple different objects present several challenges for imitation learning, with resulting policies exhibiting poor sample efficiency, generalisation, and modularity. Central to these limitations is the use of images and *absolute* coordinate systems to capture the state of the world. Without extensive demonstration datasets, these representations constrain the policy to operate over a closed set of spatial locations, intra-category instances, and even task variations. In this paper, we present a method to address these challenges using *affordance-centric* coordinate frames. By appropriately reorienting this frame and training a state-based policy using this *relative* coordinate system, we demonstrate that we can not only learn highly sample-efficient manipulation behaviours but also generalise to a wide range of spatial and intra-category object variations. More importantly, we show that this representation allows us to learn independent sub-policies that can be seamlessly composed together to solve complex, long-horizon, multi-object tasks, with the modularity for compositional generalisation to new task variations. We extensively validate our approach on a real-world tea-serving task involving 5 different objects, 13 intra-category object variations, and 7 different sub-tasks exhibiting a vast range of spatial variations, demonstrating our ability to solve the entire long-horizon task with the equivalent of only 10 demonstrations. Video demonstrations and code will be available at policy-decomposition.github.io.

Keywords: sample-efficient imitation learning, relative action frames, long-horizon manipulation

1 Introduction

To solve challenging household tasks such as cleaning dishes, serving tea, or packing away toys, robots have to directly interact with multiple objects, facilitate their interaction, and compose subtasks over relatively long time horizons. Fuelled by recent advances in policy representation [1] and data collection methods [2, 3], imitation learning promises to solve such complex tasks but still faces significant challenges in long-horizon multi-object settings with resulting policies exhibiting poor sample efficiency, generalisation, and modularity. Specifically, the combinatorial explosion in state variations associated with multiple objects requires an extensive number of demonstrations to capture the vast range of object and inter-object spatial variations. This is further exacerbated when learning policies to operate across intra-object category variations. Overall, the resulting policies exhibit limited generalisation to task variations making it difficult to compose them to solve longer horizon tasks. A key observation central to these limitations is the use of images and *absolute* coordinates to capture the state of the world.

While images provide a rich state representation, they capture a considerable amount of task-irrelevant information and require a significant number of demonstrations to cover all possible downstream variations for policy learning, resulting in poor overall generalisation. Additionally, using absolute coordinate systems for policy learning constrains the policy to operate over fixed spatial locations captured in the training data. In the multi-object setting, the dimensionality of the configuration space, representing the poses of all relevant objects, grows proportionally to the num-

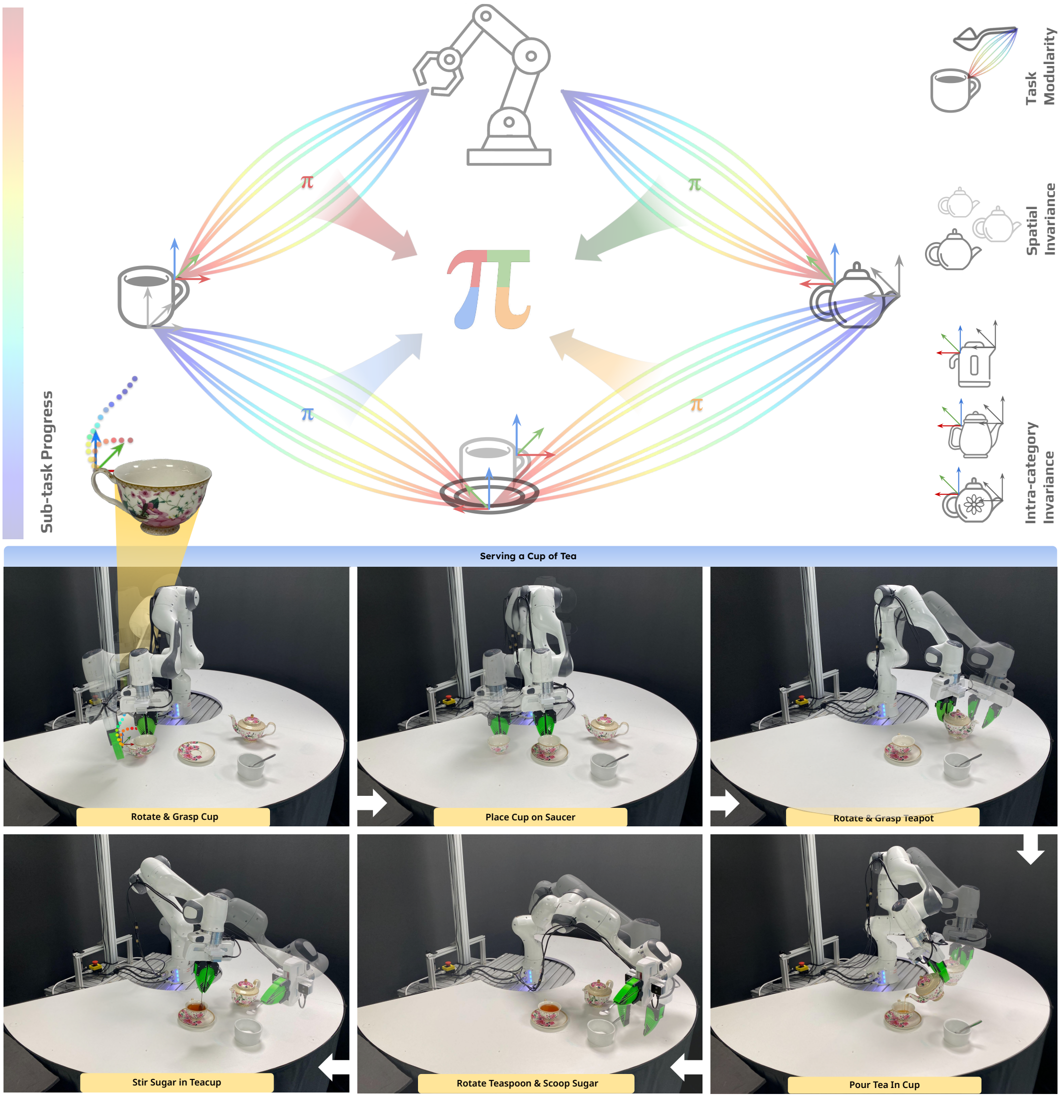


Figure 1: **Affordance-Centric Policy Decomposition** π. Our method decomposes long-horizon, multi-object policy learning into a series of sub-policies that can be trained independently using an *oriented affordance-centric* coordinate frame located on a sub-task-specific object. Each sub-policy is both spatially and intra-category invariant allowing for compositional generalisation to a wide range of variations in the downstream long-horizon task. The sub-policies are augmented to learn a notion of self-progress allowing them to autonomously switch between sub-policies without the need for a learned arbitrator.

ber of involved objects. This results in an exponential increase in the possible arrangements of the objects relative to each other and to the robot, further complicating the learning process.

Recently, a large body of work has motivated the use of local object-centric representations [4, 5] or keypoints [6], as alternative state representations to images for manipulation tasks. By placing these keypoints at specific robot-object or object-object affordance regions, these representations have been shown to enable intra-category generalisation of pick-and-place tasks using off-the-shelf inverse kinematics and motion planning algorithms. In this work, we explore how these local representations can be leveraged in imitation learning. More importantly, how these representations can enable us to learn sample efficient and generalisable diffusion policies for long-horizon, multi-object tasks.

We propose to decompose long-horizon, multi-object task learning, into a series of sub-policies each trained with respect to a *relative* coordinate frame located at task-informed affordance regions on objects. This object-centric perspective allows us to train independent sub-policies on a single object for each sub-task, circumventing the combinatorial explosion associated with multi-object tasks. For each sub-task, we reorient the affordance frame relative to the robot's tool frame, ensuring that the robot is always operating within the data support of the next sub-policy at the end of each sub-task. This, together with our augmented action space indicating self-progress, facilitates the autonomous composition of each sub-policy without requiring an additional arbitration policy – typically trained in a hierarchical imitation learning setting requiring full long-horizon task demonstrations.

The advantages of our formulation are as follows: **1)** by training the policy in a relative coordinate frame, we decouple its learned behaviours from the fixed locations seen in the training dataset allowing for spatial generalisation to new object locations, **2)** the abstraction of policy learning from images and the placement of the relative coordinate frame at the affordance centric region of objects, allows for intra-category transfer of learned behaviours, **3)** our oriented affordance frame formulation allows us to learn highly sample efficient policies from as little as 10 demonstrations and **4)** all the above abstractions allow us to collect smaller and less tedious demonstration sequences for each sub-task which can then be composed to solve longer horizon tasks.

We evaluate our proposed policy learning approach on a real-world tea-serving task involving 5 different objects, 13 intra-category object variations, and 7 different sub-tasks exhibiting a vast range of spatial variations and demonstrate our ability to solve the entire long-horizon task with the equivalent of only 10 demonstrations. Video demonstrations and code will be available on our project page[1].

2 Related Works

Long-Horizon Task Learning: Robotic manipulation has long struggled with learning long-horizon skills[7, 8, 9, 10]. By discovering and planning over sub-skills, hierarchical reinforcement learning enables exploration over extended timeframes [11, 12, 13, 14, 15, 7, 16, 17, 18]. However, the computational expense and real-world unsuitability of these algorithms often stem from needing to learn subskills and overall policies from scratch. The hierarchical imitation learning literature has seen many efforts to learn or use subgoal decomposition to break up long tasks when provided with full-task demonstrations to intermediate learning signals and mitigates compounding action errors [19, 20, 21, 22, 23, 24]. Other work propose to generate sub-goals for long-horizon tasks [10, 25, 26, 27, 28, 29], but this requires training an expensive generative model and a multi-stage approach for collecting demonstrations. Universal Visual Decomposer proposes a method for discovering sub-goals for decomposing long-horizon manipulation tasks by detecting phase shifts in pre-trained vision embeddings [30]. A key limitation of all these methods is the need for full demonstration sequences of the long-horizon task which are tedious to collect and can require extensive demonstrations to cover the vast state spaces associated with these tasks. This comes as a result of the tight coupling between the state space of the full long-horizon task, and each sub-policy. In this work, we explore how we can decouple sub-task learning without requiring full demonstration sequences of the long-horizon task by rethinking the state representation of the policy.

Object-Centric Representations for Manipulation: Raw image observations are often preferred for imitation learning because they do not depend on auxiliary perception systems, [1, 3, 31, 2] however, they induce several limitations on the generalisation of the resulting policies requiring exhaustive amounts of demonstration data. There has been a growing body of work in the manipulation literature exploring object-centric representations to address these challenges [5, 4, 32]. Object pose is typically used as state information in simulation settings however a key challenge is transferring these policies to instance variations of the object. More interestingly [6, 33, 34] have explored local object correspondences to allow for this desired intra-category invariance. These local regions are typically located around key affordance regions of an object and have been used for category invariant pick and place tasks using off-the-shelf inverse kinematics and motion planning algorithms. Pre-trained models like DINO [35, 36] have improved this correspondence matching [37] and have been used for retrieval, alignment and replay of manipulation demonstrations [38, 39]. In this work, we explore how these local affordance-centric regions can be used in the imitation learning setting, for sample efficient and generalisable policy learning of long-horizon, multi-object tasks.

Relative State and Action Spaces for Imitation Learning: In order to improve the spatial generalisation of imitation learning policies, several works have investigated the effects of using specific

[1] policy-decomposition.github.io

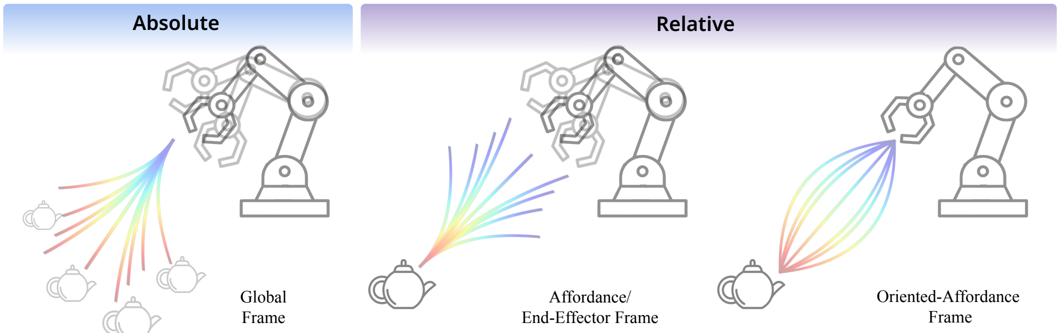


Figure 2: **Demonstration trajectory distributions for different frames of reference.** *Left:* A *fixed global* reference frame requires all spatial relative arrangements of both the end-effector and the object. *Middle:* An end-effector or affordance-centric reference frame only requires one of them to freely translate but both can freely rotate. *Right:* An *oriented*-affordance frame of reference only requires the object (or end-effector) to freely translate and rotate.

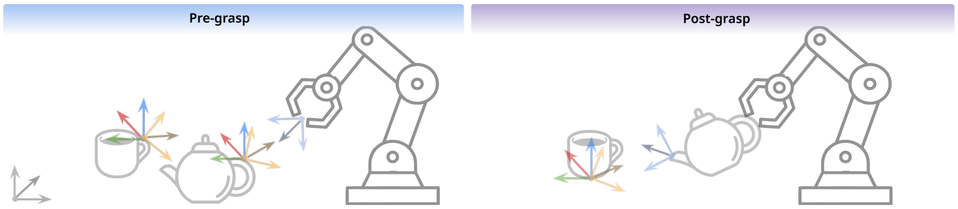


Figure 3: **Affordance Frames, Oriented-Affordance Frames and Tool Frames.** Left: Affordance frames (∟), oriented affordance frames (∟), and tool frame (∟) for a typical *pick* task. Right: Frames for the *pour* task. Notice how the oriented affordance frames are identical to the affordance frames, but rotated such that the 'funnel' axis (brown) points towards the origin of the tool frame at the beginning of the task.

coordinate systems to express a policy's state and action space. While it is common to use a fixed global frame to capture object locations, the *absolute* nature of the resulting coordinates limits generalizability. Conversely, *relative* coordinate systems have been shown to improve efficiency and generalizability as the resulting policies are no longer bound to absolute coordinates [40, 31]. In this work, we explore how expressing state and actions with respect to a local frame attached to affordance-centric regions on objects can help us decouple sub-policies from the long-horizon tasks and allow for better spatial generalisation.

3 Affordance-Centric Policy Decomposition π

3.1 Problem Formulation

Our goal is to learn a **sample efficient** and **generalisable** policy $\pi^*(a|o)$ that can generate actions a given observation o in order to solve multi-object, long horizon tasks $\mathcal{T}^*$. To simplify the learning problem, we decompose the long-horizon policy π^* into a series of sub-policies: $\{\pi_1, \pi_2, \pi_3, ...\pi_k\}$ such that: $\pi^*(a \mid o) = \pi_1(a \mid o) \circ \pi_2(a \mid o) \circ \ldots \circ \pi_k(a \mid o)$, where $\circ$ denotes the chaining of sub-policies. We aim to train independent sub-policies by collecting smaller and easier-to-collect demonstration datasets $\mathcal{D}^* = \{\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3, ...\mathcal{D}_k\}$. Our *desiderata* in this setting is for each policy to be spatially and intra-category invariant from the locations and instances that could be encountered in the downstream long horizon task. This will ensure that the resulting policies can be composed for solving extended tasks. Each sub-policy π_i is formulated as an imitation learning problem, where the goal is to learn a mapping from observations to actions based on a set of demonstrations. Formally, given a dataset of demonstrations $\mathcal{D}_i = \{(o_t, a_t)\}_{t=1}^{N_i}$ for sub-task i, where o_t is the observation at time t, and a_t is the corresponding action, we aim to minimise the discrepancy between the actions taken by the policy π_i and the actions in the demonstration dataset. The objective function for training each sub-policy π_i can be defined as $\mathcal{L}_i(\theta_i) = \mathbb{E}_{(o_t,a_t)\sim\mathcal{D}_i}\left[\|\pi_i(o_t;\theta_i) - a_t\|^2\right]$, where θ_i represents the parameters of the sub-policy π_i. The goal is to find the optimal parameters θ_i^* that minimise the objective function $\theta_i^* = \arg\min_{\theta_i} \mathcal{L}_i(\theta_i)$.

3.2 Affordance-Centric Policy Learning for Long-Horizon Tasks

Affordance Frames: The choice of reference frame for representing robot state and actions is critical, as it significantly affects a policy's ability to generalise to spatial variations in multi-object tasks. When robot poses are represented as absolute coordinates in **fixed global reference frames**, the policy's operation is limited to a closed set of spatial locations with limited generality beyond the training set. This requires the demonstrations to cover an exhaustive number of spatial locations during training as illustrated in the left panel of Fig. 2. If an object appears in a previously unseen position, the policy will be out of distribution and likely fail. Alternatively, relative coordinate systems have been shown to yield better spatial generalisation of trained policies with the appropriate placement of these frames allowing for better sample complexity as shown in the middle panel of Fig. 2. In this work, we propose to place these relative coordinate frames on the affordance-centric regions of objects. We call these task-dependent local coordinate frames *affordance frames*. Objects can have multiple task-dependent affordance frames: e.g. a cup has an affordance frame on the handle for the task of *picking up*, and an affordance frame in the centre of the cup for the task of *pouring*.

Tool Frames: In multi-object tasks, a robot would either directly interact with an object (e.g. for picking), or act on a target object while holding a *tool* object. In addition to the affordance frame defined relative to the *target* object, we define a *tool frame* on the *tool object*. For simple pickup tasks, the tool frame is identical to the robot's end-effector frame, however for actions such as stirring tea with a spoon or pouring from a teapot, the tool frame is placed on the scoop of the spoon or the sprout of the teapot respectively as illustrated in Fig. 3.

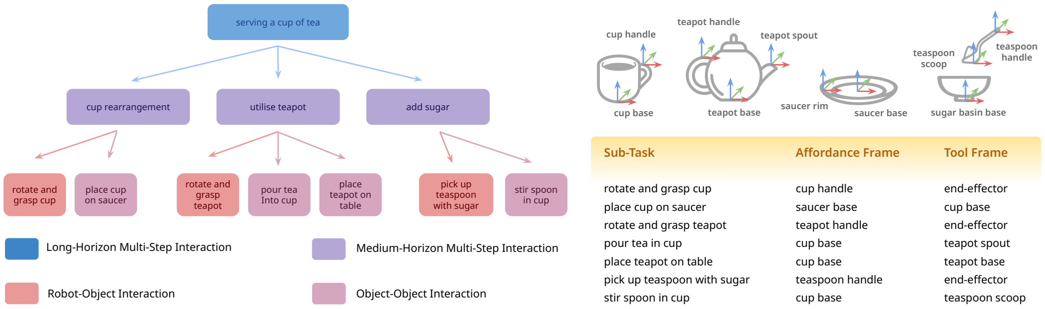
Oriented-Affordance Frames Given affordance frames and tool frames, we can now introduce the concept of the *oriented-affordance frame*. It is obtained by rotating the affordance frame on our target object such that its 'funnel' axis (in our case we chose the x-axis) is directed towards the origin of the tool frame. By defining the robot's end effector state and actions in this frame (right panel in Fig. 2), the resulting sub-policy ensures that the robot always falls within its data support despite the end-effector location or the rotation of the target object. This further simplifies data collection target object in a small set of locations with the ability to generalise to any new location in the larger workspace of the robot. More importantly, when composing subsequent policies, the oriented affordance frame ensures that the robot is always within the support of the next sub-policy at the end of a sub-task.

State and Action Representation: We define a given sub-task by its corresponding affordance and tool frame transforms as follows: $\mathcal{T}_i = \{\mathbf{T}_{\text{afford}}, \mathbf{T}_{\text{tool}}\}$ where $\mathbf{T}_{\text{afford}} \in \mathbf{SE}(3)$ and $\mathbf{T}_{\text{tool}} \in \mathbf{SE}(3)$. We can then use these transforms to obtain our oriented affordance frame $\mathbf{T}_{\text{o-aff}} \in \mathbf{SE}(3)$. We define the observation space for the corresponding sub-policy π_i as the pose of the tool-frame, ${}^{\text{o-aff}}\mathbf{T}_{\text{tool}} \in \mathbf{SE}(3)$, represented in the oriented affordance frame, and the gripper state $g_{\text{s}} \in \{0, 1\}$. The action space of the policy consists of the desired next pose of the robot's end effector ${}^{\text{o-aff}}\mathbf{T}_{\text{ee}}$ in the oriented affordance frame, and the gripper action $g_{\text{a}} \in \{0, 1\}$.

Sub-Policy Arbitration using Self-Progress: In order to autonomously chain sub-policies to solve a long-horizon task, we augment the sub-policy's action space $\mathcal{A}$ to include a task-progress indicator $\mathcal{A}^+ := \mathcal{A} \cup \{a_{\text{progress}}\}$. For each demonstration trajectory τ, we compute a task progress measure by linearly interpolating progress values from 0 to 1 based on the length of τ. The policy is then trained to output not only the action but also the corresponding task-progress value. During execution, this task-progress value is used to determine when to transition from one sub-policy to the next. Let $p_t \in [0, 1]$ denote the task progress at time step t. The action output of the sub-policy π_i is extended to include the task-progress value $\pi_i(o_t) = (a_t, p_t)$ where a_t is the action and p_t is the task-progress value. The transition between sub-policies occurs when the task-progress value p_t exceeds a predefined threshold, indicating that the current sub-task is complete and the next sub-policy should be activated. This allows us to compose sub-policies to solve complex long-horizon tasks without training an additional arbitration policy.

4 Evaluation

We systematically evaluate the various components of our affordance-centric policy decomposition framework across a complex, long-horizon, multi-object real-world manipulation task of serving a cup of tea.



Sub-Task	Affordance Frame	Tool Frame
rotate and grasp cup	cup handle	end-effector
place cup on saucer	saucer base	cup base
rotate and grasp teapot	teapot handle	end-effector
pour tea in cup	cup base	teapot spout
place teapot on table	cup base	teapot base
pick up teaspoon with sugar	teaspoon handle	end-effector
stir spoon in cup	cup base	teaspoon scoop

Figure 4: **Affordance-centric task decomposition for the tea serving task.** *Left:* Task decomposition hierarchy; *Top Right:* Affordance-centric frames for each object; *Bottom Right:* Sub-task frame definitions.

4.1 Experimental Setup

Task Description: To solve the full task of tea serving, the robot has to place a teacup on a saucer, pour tea in the teacup and then finally add sugar and stir. This task consists of 5 different objects including a teacup, saucer, teapot, sugar basin and teaspoon which can be decomposed into 7 different sub-tasks based on our affordance-centric perspective. For each object, we identify the set of affordance-centric regions which could be used across various different tasks as shown in Figure 4 (top-right). The full long-horizon task decomposition is given in Figure 4 (left) and we identify both the affordance and tool frames which define each subtask in Figure 4 (bottom-right). To thoroughly evaluate our ability to learn sample efficient and generalisable sub-policies and our ability to compose these sub-policies across varying levels of long-horizon task complexity, we conduct all evaluations across the task hierarchy shown in Figure 4.

Affordance-Centric Frame Extraction: As our system focuses on evaluating the utility of affordance-centric coordinate frames to solve long-horizon, multi-object tasks, we decouple our results from the performance of the detection system by utilising ground truth detections of these frames. We note here however that any existing perception system including [6], [33], [39] could be substituted in our framework.

Policy Training: We utilise Diffusion Policy [1] for imitation learning and train each policy for 4500 epochs with the same default parameters provided in the original implementation [1]. The state space for all the affordance-centric policies comprises a 16-D vector consisting of the 3-D position of the robot's tool frame, a 6-D representation [41] of the tool frame and object rotation, and the 1-D gripper state. For all the baselines, we additionally provide the position of the object resulting in a 19-D state vector. The action space for all methods is the same and comprises a 11-D vector consisting of the 3-D position of the robot's end effector, a 6-D representation [41] of its rotation, the 1-D gripper action and the 1-D policy's self-progress.

Baselines: We focus our evaluation on the use of the oriented affordance frame and how it compares to other frames of reference when learning sub-policies and composing them to solve long-horizon tasks. These include the default **global** reference frame with absolute coordinates and the **end-effector** relative reference frame suggested by Chi *et.al.* [31].

Evaluation Methodology: A key goal of this paper is to explore how long-horizon policy learning can be made sample efficient with the ability to generalise beyond the training data distribution. To this end, we limit our training of all sub-policies to only 10 demonstrations. This constraint allows us to better understand the generalisation capabilities of our method in the low data regime while trying to decouple its success from simple data scaling. We consider both in-domain (IND) and out-of-distribution (OOD) task evaluations. In (IND) evaluation, the policy is evaluated in regions and object arrangements it was explicitly trained on, whereas in (OOD) evaluation, we evaluate its spatial and compositional generalisation to new object and inter-object locations. For the multi-step tasks including the medium-horizon and full long-horizon tasks, we evaluate how each method's sub-policy can allow for compositionality to solve extended tasks. For each task, all objects start with the same set of initial states, matched manually with reference markings. We illustrate all the object configurations used for training and evaluation in Figure 9.

4.2 Experimental Results

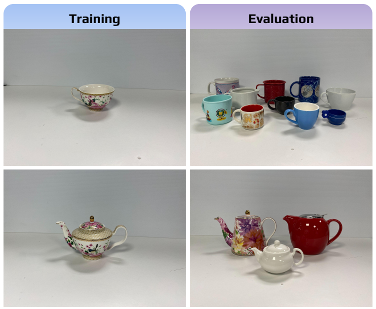
We summarise our evaluation of each sub-policy in Table 1 and our compositional evaluation when solving extended tasks in Table 2. Across all evaluations, our oriented affordance frame consistently outperforms all alternative methods across both learning individual sub-tasks and when composing these policies to solve long horizon tasks with an average success rate of 83.1% in the (OOD) for each individual sub-policy and 70.2% in the compositional setting. We note here that all evaluations are focused on the low data regime with each policy only trained with 10 demonstrations. More importantly, we show in Figure 11 (c) that the overall performance of our method could be significantly increased by increasing the number of demonstrations to $>$30.

4.2.1 Key Findings

Oriented affordance frames enables sample efficiency and better spatial generalisation. We found that the oriented frame plays an important role in enabling sample-efficient learning of each individual sub-policy, significantly outperforming all baselines in the (IND) setting. While both the end effector frame and oriented affordance frame can enable the sub-policy to generalise beyond the spatial variations captured by the demonstrations in the (OOD) setting, we note that our method still attains a higher performance. We attribute this to two main reasons: 1) Firstly the oriented frame allows use to maximise the efficacy of the limited 10 demonstrations by aligning the data support of the policy towards the tool frame of the robot, ensuring that it is always in-distribution with respect to this relative frame and 2) without the anchored oriented frame, the robot using the end-effector frame tended to constantly violate joint limits – we discuss this in detail below.

As shown in Figure 11(b) we contrast the ability of our method to learn a spatially invariant policy from the equivalent of just 10 demonstrations for the cup rearrangement task when compared to a similar image-based policy [31] which required training on 305 demonstrations[2].

Reduced joint limit violations While relative action frames provide the ability to generalize policies to new spatial configurations, we found that the end effector might rotate to achieve a desired pose relative to the object without considering the robot's base orientation. This often led to awkward and constrained configurations, resulting in joint limit violations. These violations were a common occurrence that led to failed trials when evaluating the end-effector-centric baseline, particularly in tasks where the robot needed to rotate the object. A similar issue arose when we ablated the need for the oriented frame, as shown in Figure 11 (left). Our key insight from these observations is the critical importance of the oriented affordance frame. By anchoring the movements of the end-effector relative to the object's affordance frame, the robot can avoid excessive rotations and awkward configurations.



Task	# Instances	Success
cup rearrange	10	8/10
utilise teapot	3	3/3

Figure 5: **Generalisation to intra-category variations** The set of objects used for training and evaluating the intra-category generalisation capabilities of the trained sub-policies.

Compositional generalisation to new object arrangements. An important objective of this research was to develop independent sub-policies that could effectively be composed to solve extended tasks involving multiple objects. Across both in-domain (IND) and out-of-distribution (OOD) settings, our approach consistently outperforms all baselines, as summarised in Table 2. This is a challenging task as it requires each preceding sub-task to perform successfully in order for the subsequent sub-tasks to continue. We can attribute the high success rate of our method to the oriented affordance frame which played a crucial role in ensuring that the robot's tool frame always landed within the support of the next sub-policy, ensuring seamless composition and operation of each subsequent policy.

Intra-category invariance for imitation learning By attaching our relative frame for imitation learning at the affordance-centric regions of an object, we gain the ability to transfer our trained policy across a wide range of intra-category variations. We illustrate this in Figure 5 where we train

[2] We attempted to follow a similar evaluation protocol as per [31] with a broader set of spatial variations.

		Oriented Affordance Frame (Ours)		End Effector Frame		Global Frame	
Task	**# of Demos**	**IND Success**	**OOD Success**	**IND Success**	**OOD Success**	**IND Success**	**OOD Success**
rotate and grasp cup	10	81.8%	81.8%	45.5%	45.5%	45.5%	0.0%
place cup on saucer	10	100%	100%	100%	100%	9.1%	0.0%
rotate and grasp teapot	10	90.9%	81.8%	27.3%	27.3%	81.8%	0.0%
pour tea into cup	10	100%	81.8%	45.5%	27.3%	54.5%	0.0%
place teapot on table	10	90.9%	72.7%	54.5%	54.5%	90.9%	0.0%
pick up teaspoon with sugar	10	81.8%	81.8%	45.5%	27.3%	72.7%	0.0%
stir spoon in cup	10	90.9%	81.8%	18.2%	9.1%	72.7%	0.0%

Table 1: **Sub-policy evaluation.** Success rate across both in-distribution (IND) and out-of-distribution (OOD) spatial configurations of objects for each sub-task.

		Oriented Affordance Frame (Ours) (Composition)			Global Frame (Composition)			End Effector Frame (Composition)		
Task	**# of Demos**	**State Dim**	**IND Success**	**OOD Success**	**State Dim**	**IND Success**	**OOD Success**	**State Dim**	**IND Success**	**OOD Success**
cup rearrangement	10	16	81.8%	81.8%	19	36.4%	0.0%	19	45.5%	0.0%
utilise teapot	10	16	81.8%	63.3%	19	45.5%	0.0%	19	18.2	9.1%
add sugar	10	16	72.2%	72.2%	19	63.6%	0.0%	19	9.1%	9.1%
serving a cup of tea	10	16	81.8%	63.3%	19	0.0%	0.0%	19	9.1%	9.1%

Table 2: **Policy compositionality evaluation.** Success rate across both in-distribution (IND) and out-of-distribution (OOD) spatial configurations of multiple objects for compositional tasks.

both the cup rearrangement and teapot utilisation tasks on a single cup and teapot set as shown in the right panels. The same trained policy was evaluated across a wide range of variations ranging from colour, shape and size, with each policy achieving almost a perfect success rate. Allowing the tool-frame state of the policy to vary based on the object's shape played an important role in generalising the policy to larger intra-category variations where the cup was significantly smaller or the spout of the tea-pot retracted significantly as shown in the bottom right of Figure 5.

5 Limitations and Conclusion

In this paper, we introduced a novel method for factorising robot policy learning into independent, affordance-centric sub-policies, aimed at addressing the challenges of sample efficiency and generalizability in long-horizon, multi-object manipulation tasks. By leveraging oriented affordance frames, our approach enables robots to learn complex tasks from a minimal number of demonstrations, generalise to new spatial configurations, and handle intra-category object variations effectively. Our experimental results validate the effectiveness of our method, demonstrating substantial improvements in both task performance and spatial generalisation compared to traditional global and end-effector frames. The ability to seamlessly compose sub-policies to solve long-horizon tasks without the need for exhaustive full-task demonstrations marks a significant step forward in imitation learning for robotic manipulation.

While our approach is effective for learning long-horizon robotics tasks, we note several limitations that warrant further exploration. Firstly, the current approach depends on reliable category-level tracking for continually obtaining the affordance frames during a manipulation task. While a variety of promising solutions exist for tracking even through occlusions [42, 43, 44], this is still an open research area. Additionally, this makes our approach not directly applicable to non-rigid bodies, where alternative state representations might be required. Moreover, the pose-based abstraction of observations could limit the applicability to tasks not easily represented by object affordance frames alone, potentially requiring additional modalities like tactile sensing. Despite these limitations, our contribution offers a promising pathway to more sample-efficient and generalisable imitation learning of complex long-horizon manipulation tasks.

Supplementary Material

A Implications of Affordance-Centric Imitation Learning

A.1 Closed Loop Control

Prior work have introduced the concept of local keypoints or regions on objects as compact representations for manipulation tasks [33, 6, 39]. These systems have traditionally been used to define start and end poses for simple pick-and-place operations, utilizing off-the-shelf inverse kinematics and motion planners to move objects from one location to another in an open-loop manner [33, 6]. Other methods have incorporated these representations within the context of imitation learning, primarily focusing on one-shot imitation learning [45, 46, 47]. In these cases, the keypoint locations are used to define complex admittance controllers [45] or prompt large language model (LLM) [39] to replicate a single trajectory, limiting their ability to react to changes or perturbations during policy execution. Our approach in contrast, leverages these representations in a behaviour cloning setting where we can learn closed-loop diffusion policies [1] that are robust to perturbations and allow us to move beyond simple pick-and-place tasks to imitating more complex closed-loop tasks, including non-prehensile manipulation, such as pushing objects as shown in Figure 6 below.

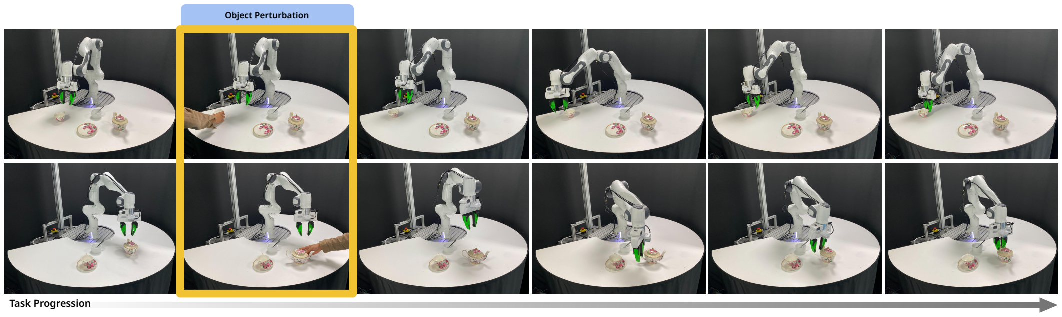


Figure 6: **Closed-loop control.** By extending the use of these affordance-centric keypoints to the behaviour cloning setting we demonstrate our ability to learn robust policies that can react to object perturbations during execution, with applicability to learning tasks beyond simple pick-and-place.

A.2 Intra-Category Generalisation

We provide additional qualitative results to demonstrate the ability of our trained affordance-centric policy to adapt to a wide range of intra-category variations, specifically size and form factor as shown in Figure 7. A key component to this generalisation comes from our task-informed use of object affordances to define task-axes and tool-frames of the robot. For robot-object interaction tasks such as rotating and grasping a cup, the placement of the relative affordance frame at the point of interaction on the object (e.g. handle) allows the resulting policy to generalise across most intra-object variations involving handles (e.g. mugs, teacups) as shown in the bottom row of Figure 7. Once the robot grasps an object, a new set of affordances becomes available, enabling the robot to perform different object-object tasks such as placing a cup on a saucer or pouring tea into a cup where the specific task-relevant affordance for this object shifts from the handle to other parts of the object, such as the base of the cup or spout of the teapot respectively. Re-targeting the tool-frame of the robot to these task-informed affordances enables the policy to generalise to large shape and size variations in object-object interaction tasks as demonstrated in Figure 8.

A.3 Spatial Generalisation Evaluation

All sub-policies were trained with a demonstration dataset collected within a small fixed set of spatial locations allowing us to capture trajectory variations, recovery behaviours and inter-object variations important for learning the sub-task using behaviour cloning. These spatial locations are illustrated in the *Training* column of Figure 9. For the OOD evaluation of our system discussed in Section 4, we expand the potential locations of the objects across a broader range of unseen locations as shown in the *Evaluation* column of Figure 9. As we train our policy within a relative task coordinate frame

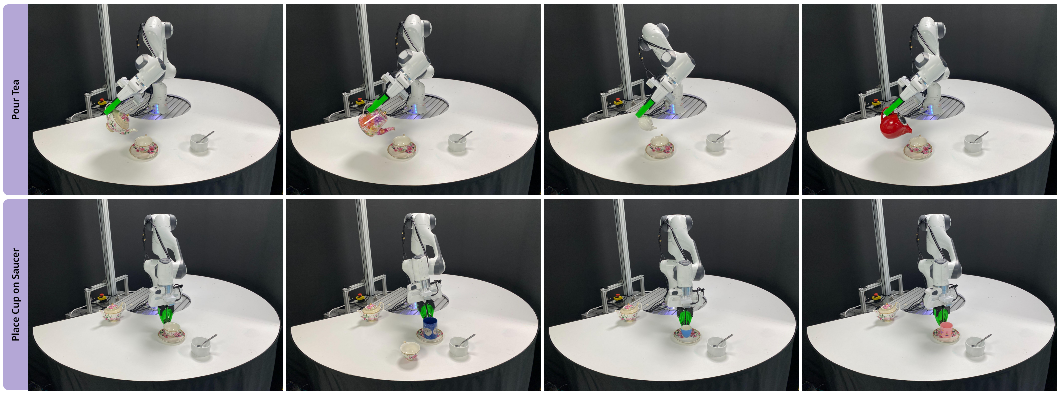


Figure 7: **Intra-category generalisation.** We demonstrate the ability of our approach to enable generalisation across large shape and size intra-category variations.

that is located on the object, we gain spatial invariance to the location of the object. Note how these smaller and simpler-to-collect sub-task demonstrations enable us to learn complex multiple-object compositional policies that can operate over a vast range of inter-object spatial variations as shown in the *Long-Horizon Multi Task* evaluation panel of Figure 9.

A.4 Applicability to Mobile Manipulation

By training our policy with respect to a relative frame attached to an object, the robot's action and state space remain consistent regardless of the position of the robot's base. This allows for the policy to continue operation while the base of the robot is in motion. We demonstrate this by running the same policy trained in the tabletop setting on a mobile manipulator robot and show how the end effector of the robot can maintain task performance regardless of the movement of the robot's base as illustrated by the discrepancy between the green and red robot base locations in Figure 10.

B Analysis of Coordinate Frames for Imitation Learning

In this section, we provide an analysis of the different coordinate frames explored in this work for imitation learning, and their state space complexities.

B.1 Global Frame

Definition. The global frame represents both the end-effector and object positions and orientations in a fixed, absolute coordinate system. This frame captures the complete spatial configuration of the robot and the N objects in the workspace.

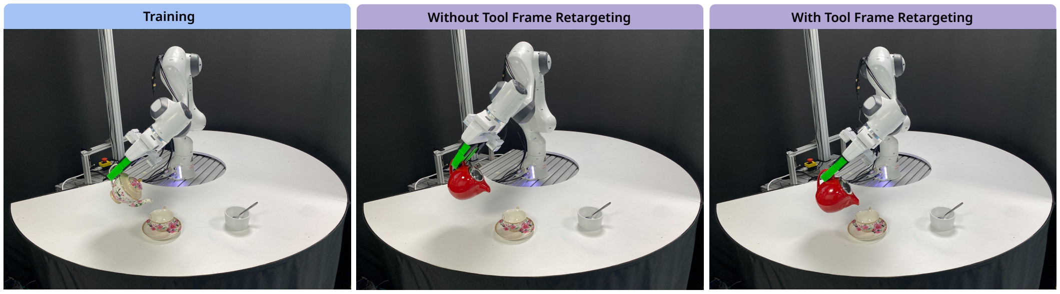


Figure 8: **Tool-Frame Re-targeting.** We illustrate the importance of tool frame re-targeting to the spout of the teapot to enable large intra-category variations based on the spout location.

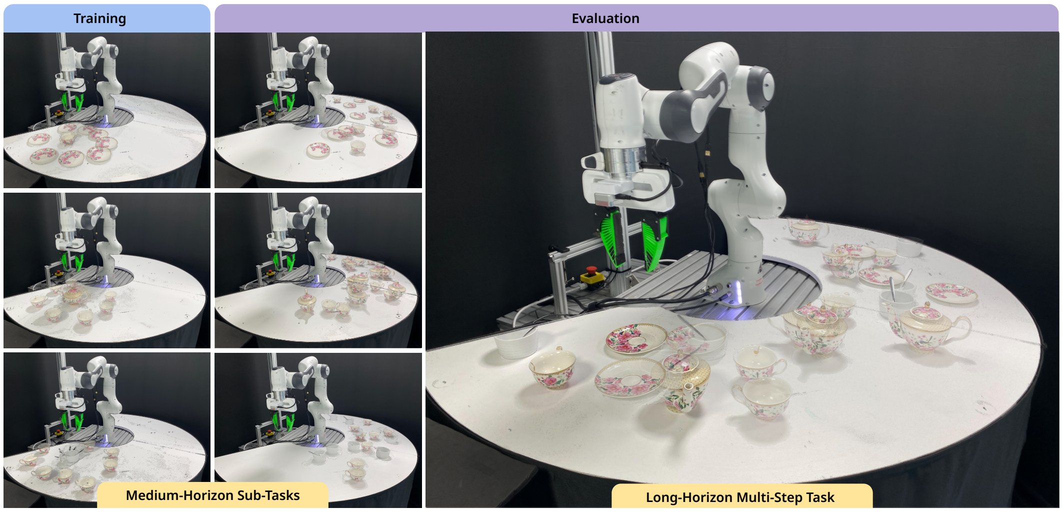


Figure 9: **Training and OOD Evaluation Object Start Configurations.** Spatial start configurations of objects across tasks used for training and evaluation in the out-of-distribution setting.

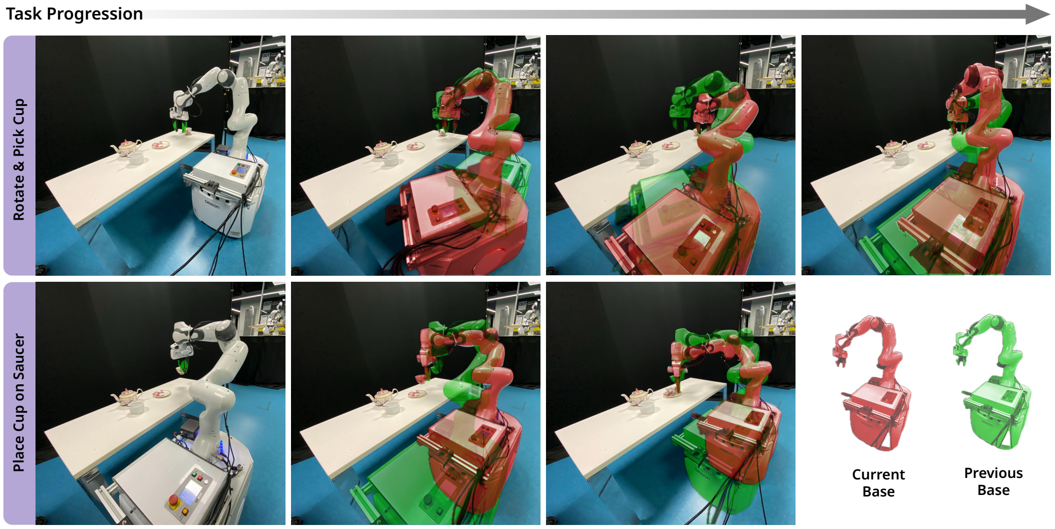


Figure 10: **Robustness to moving base.** We demonstrate our ability to maintain task performance regardless of the robot's moving base when operating with respect to an affordance-centric task frame.

State Space:

$$\mathcal{S}_{\text{global}} = \{(\mathbf{T}_{\text{tool}}, \mathbf{T}_{\text{o}1}, \mathbf{T}_{\text{o}2}, \dots, \mathbf{T}_{\text{oN}})\}$$

where

$$\mathbf{T}_{\text{tool}} = \begin{pmatrix} \mathbf{R}_{\text{tool}} & \mathbf{t}_{\text{tool}} \\ 0 & 1 \end{pmatrix}, \quad \mathbf{T}_{\text{o}i} = \begin{pmatrix} \mathbf{R}_{\text{o}i} & \mathbf{t}_{\text{o}i} \\ 0 & 1 \end{pmatrix} \quad \text{for } i = 1, 2, \dots, N$$

$\mathbf{T}_{\text{tool}}, \mathbf{T}_{\text{o}i} \in \mathbf{SE}(3), \mathbf{R}_{\text{tool}}, \mathbf{R}_{\text{o}i} \in \mathbf{SO}(3)$ and $\mathbf{t}_{\text{tool}}, \mathbf{t}_{\text{o}i} \in \mathbb{R}^3$.

The dimensionality of this state space is the sum of the translational $(\mathbf{t}_{\text{tool}}, \mathbf{t}_{\text{o}i})$ and rotational $(\mathbf{R}_{\text{tool}}, \mathbf{R}_{\text{o}i})$ components for both the end-effector tool frame and all N objects and encapsulates the various degrees-of-freedom that have to be covered by the demonstrations:

$$\dim(\mathcal{S}_{\text{global}}) = 6 + 6N$$

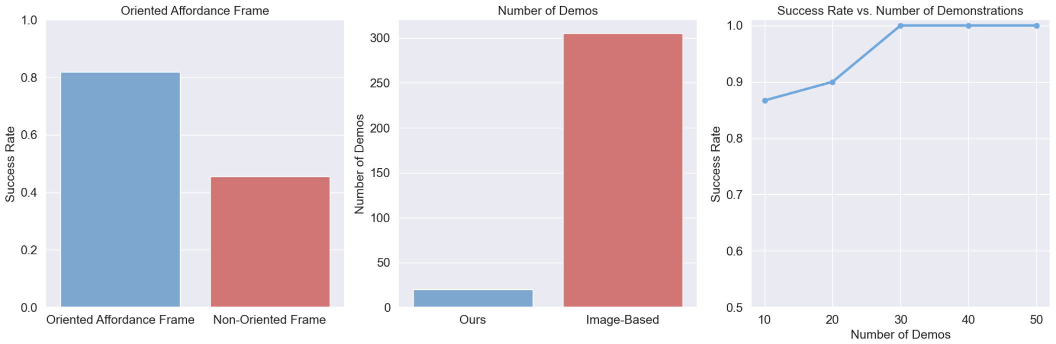


Figure 11: **Additional Comparisons.** *Left:* Oriented affordance frame vs. affordance frame; *Middle:* Relative number of demonstrations required for standard image-based diffusion policy[31] vs. Oriented affordance frame on the cup rearrangement task; *Right:* Success rate vs. Number of demonstrations on the cup rearrangement task.

B.2 Affordance Frame

Definition. The affordance frame $\mathbf{T}_{\text{afford}} \in \mathbf{SE}(3)$ is a local frame attached to an object at a location corresponding to its task-specific utility. When training policies within this frame, we only require to capture the space corresponding to the relative position and orientation of the end-effector tool frame with respect to this affordance-centric region. Training a policy using this relative coordinate system as the origin provides it with spatial invariance to the location of the object, reducing the effective space that the demonstrations have to cover to a local volume around the object. As each affordance frame corresponds to a different object interaction, we decompose task learning into a series of affordance-centric sub-tasks, significantly reducing the required state space coverage of each sub-policy by a factor of N, simplifying data collection.

State Space:

$$\mathcal{S}_{\text{afford}} = \{{}^{\text{afford}}\mathbf{T}_{\text{tool}}\}$$

where

$${}^{\text{afford}}\mathbf{T}_{\text{tool}} = \mathbf{T}_{\text{afford}}^{-1}\mathbf{T}_{\text{tool}} = \begin{pmatrix} {}^{\text{afford}}\mathbf{R}_{\text{tool}} & {}^{\text{afford}}\mathbf{t}_{\text{tool}} \\ 0 & 1 \end{pmatrix}$$

The dimensionality of this state space includes the translation and rotation of the end-effector relative to the affordance frame corresponding to the subtask. This is a further reduction from the global state space dimension by the 6 dimensions corresponding to the spatial location of the object:

$$\dim(\mathcal{S}_{\text{afford}}) = 6$$

B.3 Oriented Affordance Frame

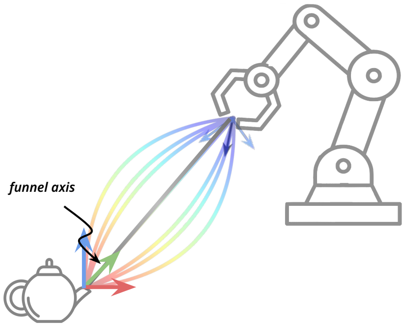


Figure 12: **Funnel Axis.** Illustration of the funnel axis and how all the demonstrations are densely concentrated along this axis.

Definition. The oriented affordance frame $\mathbf{T}_{\text{o-aff}} \in \mathbf{SE}(3)$ is similar to the affordance frame described above but is oriented such that its 'funnel' axis (Figure 12) is aligned to point towards the robot's tool frame at the start of each sub-task. This alignment normalises the state space used for policy learning, simplifying the mapping function that the policy has to learn and enhancing its ability to compose sub-tasks. The alignment forces all demonstrations to concentrate along a known axis which increases the data support from the limited number of demonstrations within a known volume. During inference, by aligning this axis with the robot's tool frame at the start of each sub-task, we facilitate the robot to seamlessly continue operation from the previous sub-policy, by ensuring that the end-effector always falls within this known volume of demonstrations. Furthermore, the fixed

Frame	State Representation	Dimensionality
Global Frame	$s = (\mathbf{T}_{\text{tool}}, \mathbf{T}_{\text{o1}}, \mathbf{T}_{\text{o2}}, \ldots, \mathbf{T}_{\text{oN}})$	$6 + 6N$
Affordance Frame	$s = {}^{\text{afford}}\mathbf{T}_{\text{tool}}$	6
Oriented Affordance Frame	$s = {}^{\text{o-aff}}\mathbf{T}_{\text{tool}}$	6

Table 3: **State-Space Complexity.** Summary of state space complexities for the different coordinate frames explored in this work.

rotation of this relative frame acts as an anchor for the rotations of the robot's end effector reducing kinematic joint violations when acting with respect to a relative task axes as discussed in Section 4.

State Space:

At the start of each episode, we perform an affine transformation to reorient the affordance frame $\mathbf{T}_{\text{afford}}$ such that its 'funnel' axis is aligned with the end-effector tool frame $\mathbf{T}_{\text{tool}}$. To achieve this reorientation, we apply a rotation $\mathbf{R}_{\text{align}}$ (Algorithm 1) to $\mathbf{T}_{\text{afford}}$:

$$\mathbf{T}_{\text{o-aff}} = \mathbf{R}_{\text{align}} \mathbf{T}_{\text{afford}}$$

This allows us to define our oriented affordance state space as:

$$\mathcal{S}_{\text{o-aff}} = \{{}^{\text{o-aff}}\mathbf{T}_{\text{tool}}\}$$

where

$${}^{\text{o-aff}}\mathbf{T}_{\text{tool}} = \mathbf{T}_{\text{o-aff}}^{-1} \mathbf{T}_{\text{tool}} = \begin{pmatrix} {}^{\text{o-aff}}\mathbf{R}_{\text{tool}} & {}^{\text{o-aff}}\mathbf{t}_{\text{tool}} \\ 0 & 1 \end{pmatrix}$$

The dimensionality of this state space is the same as in the affordance frame however this simple augmentation allows us to control how demonstrations are distributed across this space, allowing us to maximise the utility of as little as 10 demonstrations:

$$\dim(\mathcal{S}_{\text{o-aff}}) = 6$$

C Implementation Details

C.1 Experimental Setup

For all experiments, we utilise a Franka Panda manipulator arm equipped with two D405 Intel Realsense cameras attached to the flange as shown in Figure 13. We utilised Cartesian impedance control to control the robot with all demonstrations collected using a GELLO teleoperation device [48]. During data collection, all objects are equipped with an independent ArUco marker (Figure 13) which we use for identifying the affordance-centric frames via measured rigid transforms from this marker as well as for tracking these frames across the demonstration. We chose this method to obtain the affordance frame as it allowed us to decouple the performance of the perception system from the utility of the affordance frames for policy learning and composition which was the main focus of this work. While we did explore the use of existing methods [49, 6, 33] to acquire these affordance-centric regions we found that they struggle with tracking objects, particularly in closed-loop tasks involving non-prehensile manipulation. Concurrent work has been exploring the development of a dense keypoint level detection system based on [44] with the ability to track these keypoints in real-time regardless of occlusions and we demonstrate an integration of these methods in the videos on our project page[3].

[3] policy-decomposition.github.io

Subtask	Ctrl	To	Ta	Tp	#D-Params	Lr	WDecay	Symm	Do	Da
Rotate and Grasp Cup	Pos	2	8	16	8.08	1e-4	1e-6	No	16	11
Place Cup on Saucer	Pos	2	8	16	8.08	1e-4	1e-6	Yes	10	11
Rotate and Grasp Teapot	Pos	2	8	16	8.08	1e-4	1e-6	No	16	11
Pour Tea Into Cup	Pos	2	8	16	8.08	1e-4	1e-6	Yes	10	11
Place Teapot on Table	Pos	2	8	16	8.08	1e-4	1e-6	Yes	10	11
Pick Up Teaspoon with Sugar	Pos	2	8	16	8.08	1e-4	1e-6	No	16	11
Stir Spoon in Cup	Pos	2	8	16	8.08	1e-4	1e-6	Yes	10	11

Table 4: **Hyperparameters for each subtask in the tea-serving task. Ctrl**: position or velocity control. **To**: observation horizon. **Ta**: action horizon. **Tp**: action prediction horizon. **#D-Params**: diffusion network number of parameters in millions. **Lr**: learning rate. **WDecay**: weight decay. **Symm**: if the affordance frame is at a location of symmetry on the object. **Do**: dimension of the observation input vector. **Da**: dimension of the action output vector.

C.2 Policy Composition

Having trained each affordance-centric policy, we can compose them to solve long-horizon, multi-object tasks. We first define the order of sub-tasks and their associated affordance frames required to complete the full task. The robot then performs an initialisation scan of the environment to identify the initial locations of all objects and their local affordance frames in the scene. Once identified it runs the first policy corresponding to the first sub-task. As the policy is trained to output end-effector poses defined in the oriented affordance frame ${}^{\text{o-aff}}\mathbf{T}_{\text{ee}}$, we transform these actions to the base frame of the robot $\mathbf{T}_{\text{ee}}$ before executing them with a Cartesian impedance controller. If a_{progress} generated by the policy increases beyond a predefined threshold ϕ, indicating sub-task completion, we switch affordance frames and repeat the process with the next policy corresponding to the next sub-task.

C.3 Diffusion Policy

Throughout this work, we leverage diffusion policies [1] as our central behaviour cloning algorithm. Diffusion policy models the conditional action distribution as a denoising diffusion probabilistic model (DDPM), allowing for better representation of the multi-modality in human-collected demonstrations. Specifically, diffusion policy uses DDPM to model the action sequence $p(\mathbf{A}_t \mid \mathbf{o}_t, \mathbf{x}_t)$, where $\mathbf{A}_t = \{\mathbf{a}_t, \dots, \mathbf{a}_{t+C}\}$ represents a chunk of next C actions. The final action is output of the following denoising process:

$$\mathbf{A}_t^{k-1} = \alpha(\mathbf{A}_t^k - \gamma\epsilon_\theta(\mathbf{o}_t, \mathbf{x}_t, \mathbf{A}_t^k)) + \mathcal{N}(0, \sigma^2\mathbf{I}), \quad (1)$$

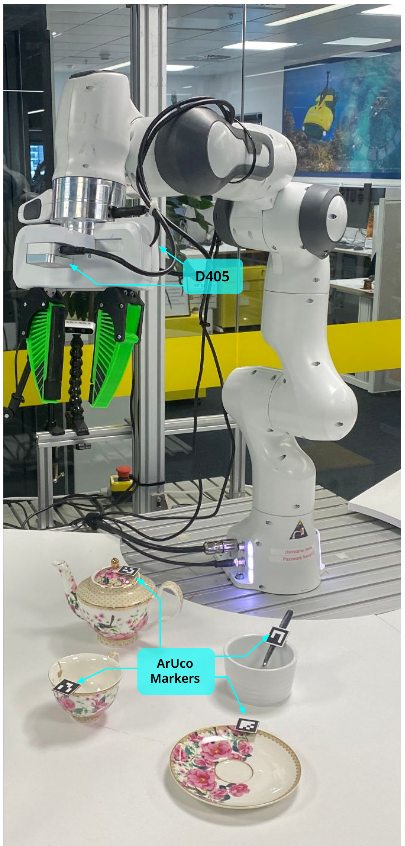


Figure 13: **Experimental Setup.** Franka Panda manipulator robot equipped with 2 D405 Realsense cameras. Objects with ArUco markers for affordance-centric frame detection and tracking.

where $\mathbf{A}_t^k$ is the denoised action sequence at time k. Denoising starts from $\mathbf{A}_t^K$ sampled from Gaussian noise and is repeated till $k = 1$. In Equation (1), (α, γ, σ) are the parameters of the denoising process and ϵ_θ is the score function trained using the MSE loss $\ell(\theta) = (\epsilon_k - \epsilon_\theta(\mathbf{o}_t, \mathbf{x}_t, \mathbf{A}_t^k + \epsilon_k))^2$. The noise at step k of the diffusion process, ϵ_k, is sampled from a Gaussian of appropriate variance. We provide the important hyperparameters used for training the policy across each sub-task in Table 4 with the rest of the implementation identical to the original implementation [1].

Algorithm 1: Calculation of R_{align}

```
Input: $\mathbf{p}_{\text{tool}}, \mathbf{p}_{\text{afford}}$
Output: $\mathbf{R}_{\text{align}}$
Function ComputeRotationMatrix($\mathbf{p}_{tool}, \mathbf{p}_{afford}$):
    Define the Vectors:
    $\mathbf{v}_{\text{funnel}} \leftarrow [1, 0, 0]^T$
    $\mathbf{p}_{\text{tool}} \leftarrow$ Position of the tool frame
    $\mathbf{p}_{\text{afford}} \leftarrow$ Position of the affordance frame
    Calculate the Direction Vector:
    $\mathbf{d} \leftarrow \mathbf{p}_{\text{tool}} - \mathbf{p}_{\text{afford}}$
    $\mathbf{d}_{\text{norm}} \leftarrow \frac{\mathbf{d}}{\|\mathbf{d}\|}$
    Find the Rotation Axis and Angle:
    $\mathbf{r} \leftarrow \mathbf{v}_{\text{funnel}} \times \mathbf{d}_{\text{norm}}$
    $\mathbf{r}_{\text{norm}} \leftarrow \frac{\mathbf{r}}{\|\mathbf{r}\|}$
    $\cos(\theta) \leftarrow \mathbf{v}_{\text{funnel}} \cdot \mathbf{d}_{\text{norm}}$
    $\sin(\theta) \leftarrow \|\mathbf{r}\|$
    Construct the Rotation Matrix:
    $\mathbf{K} \leftarrow \begin{bmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{bmatrix}$
    $\mathbf{R}_{\text{align}} \leftarrow I + \sin(\theta)\mathbf{K} + (1 - \cos(\theta))\mathbf{K}^2$
    return $\mathbf{R}_{align}$
```

Algorithm 2: Policy Composition for Long Horizon Tasks

```
Given: Initial state $s_0$, Subpolicies $\pi_1, \pi_2, \ldots, \pi_n$, Threshold $\phi$
Input: $s_0, \pi_1, \pi_2, \ldots, \pi_n, \phi$
Output: Action sequence $\{a_1, a_2, \ldots, a_T\}$
Detect all affordance frames                              ▷ Identify affordance frames
$s \leftarrow s_0$                                        ▷ Initialize state
$A \leftarrow \{\}$                                       ▷ Initialize action sequence
$i \leftarrow 1$                                          ▷ Set current subpolicy
while $i \leq n$ do
    Orient $\mathbf{T}_{\text{o-aff}}$ towards $\mathbf{T}_{\text{tool}}$          ▷ Align affordance frame
    $s_i \leftarrow \mathbf{T}_{\text{o-aff}}^{-1}\mathbf{T}_{\text{tool}}$        ▷ Transform tool frame
    while Progress $p \leq \phi$ do
        $(a_{\text{o-aff}}, p) \leftarrow \pi_i(s_i)$                ▷ Generate action and progress
        $a \leftarrow \mathbf{T}_{\text{o-aff}} a_{\text{o-aff}}$    ▷ Transform action to base frame
        Execute $a$                                       ▷ Perform action
        $s \leftarrow$ Observe new state                  ▷ Update state
        Append $a$ to $A$                                 ▷ Record action
        Orient $\mathbf{T}_{\text{o-aff}}$ towards $\mathbf{T}_{\text{tool}}$      ▷ Realign frame
        $s_i \leftarrow \mathbf{T}_{\text{o-aff}}^{-1}\mathbf{T}_{\text{tool}}$    ▷ Update state input
    $i \leftarrow i + 1$                                  ▷ Next subpolicy
return $A$                                                ▷ Return action sequence
```

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] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023.

[3] Z. Fu, T. Z. Zhao, and C. Finn. Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation. In *arXiv*, 2024.

[4] Y. Zhu, A. Joshi, P. Stone, and Y. Zhu. Viola: Object-centric imitation learning for vision-based robot manipulation. In *6th Annual Conference on Robot Learning*, 2022.

[5] S. Rezapour Lakani, A. J. Rodríguez-Sánchez, and J. Piater. Towards affordance detection for robot manipulation using affordance for parts and parts for affordance. *Autonomous Robots*, 43:1155–1172, 2019.

[6] L. Manuelli, W. Gao, P. Florence, and R. Tedrake. kpam: Keypoint affordances for category-level robotic manipulation. In *The International Symposium of Robotics Research*, pages 132–157. Springer, 2019.

[7] A. Gupta, V. Kumar, C. Lynch, S. Levine, and K. Hausman. Relay policy learning: Solving long-horizon tasks via imitation and reinforcement learning. *arXiv preprint arXiv:1910.11956*, 2019.

[8] M. Heo, Y. Lee, D. Lee, and J. J. Lim. Furniturebench: Reproducible real-world benchmark for long-horizon complex manipulation. *arXiv preprint arXiv:2305.12821*, 2023.

[9] Y. Lee, E. S. Hu, and J. J. Lim. Ikea furniture assembly environment for long-horizon complex manipulation tasks. In *2021 ieee international conference on robotics and automation (icra)*, pages 6343–6349. IEEE, 2021.

[10] A. Mandlekar, D. Xu, R. Martín-Martín, S. Savarese, and L. Fei-Fei. Learning to generalize across long-horizon tasks from human demonstrations. *arXiv preprint arXiv:2003.06085*, 2020.

[11] P.-L. Bacon, J. Harb, and D. Precup. The option-critic architecture. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017.

[12] A. Bagaria and G. Konidaris. Option discovery using deep skill chaining. In *International Conference on Learning Representations*, 2019.

[13] E. Chane-Sane, C. Schmid, and I. Laptev. Goal-conditioned reinforcement learning with imagined subgoals. In *International Conference on Machine Learning*, pages 1430–1440. PMLR, 2021.

[14] J. Co-Reyes, Y. Liu, A. Gupta, B. Eysenbach, P. Abbeel, and S. Levine. Self-consistent trajectory autoencoder: Hierarchical reinforcement learning with trajectory embeddings. In *International conference on machine learning*, pages 1009–1018. PMLR, 2018.

[15] B. Eysenbach, R. R. Salakhutdinov, and S. Levine. Search on the replay buffer: Bridging planning and reinforcement learning. *Advances in neural information processing systems*, 32, 2019.

[16] O. Nachum, S. Gu, H. Lee, and S. Levine. Near-optimal representation learning for hierarchical reinforcement learning. *arXiv preprint arXiv:1810.01257*, 2018.

[17] O. Nachum, S. S. Gu, H. Lee, and S. Levine. Data-efficient hierarchical reinforcement learning. *Advances in neural information processing systems*, 31, 2018.

[18] L. Zhang, G. Yang, and B. C. Stadie. World model as a graph: Learning latent landmarks for planning. In *International conference on machine learning*, pages 12611–12620. PMLR, 2021.

[19] J. Borja-Diaz, O. Mees, G. Kalweit, L. Hermann, J. Boedecker, and W. Burgard. Affordance learning from play for sample-efficient policy learning. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 6372–6378. IEEE, 2022.

[20] D.-A. Huang, S. Nair, D. Xu, Y. Zhu, A. Garg, L. Fei-Fei, S. Savarese, and J. C. Niebles. Neural task graphs: Generalizing to unseen tasks from a single video demonstration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8565–8574, 2019.

[21] S. James and A. J. Davison. Q-attention: Enabling efficient learning for vision-based robotic manipulation. *IEEE Robotics and Automation Letters*, 7(2):1612–1619, 2022.

[22] L. X. Shi, A. Sharma, T. Z. Zhao, and C. Finn. Waypoint-based imitation learning for robotic manipulation. *arXiv preprint arXiv:2307.14326*, 2023.

[23] M. Shridhar, L. Manuelli, and D. Fox. Perceiver-actor: A multi-task transformer for robotic manipulation. In *Conference on Robot Learning*, pages 785–799. PMLR, 2023.

[24] D. Xu, S. Nair, Y. Zhu, J. Gao, A. Garg, L. Fei-Fei, and S. Savarese. Neural task programming: Learning to generalize across hierarchical tasks. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 3795–3802. IEEE, 2018.

[25] K. Fang, P. Yin, A. Nair, and S. Levine. Planning to practice: Efficient online fine-tuning by composing goals in latent space. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4076–4083. IEEE, 2022.

[26] K. Fang, P. Yin, A. Nair, H. R. Walke, G. Yan, and S. Levine. Generalization with lossy affordances: Leveraging broad offline data for learning visuomotor tasks. In *Conference on Robot Learning*, pages 106–117. PMLR, 2023.

[27] D. Jayaraman, F. Ebert, A. A. Efros, and S. Levine. Time-agnostic prediction: Predicting predictable video frames. *arXiv preprint arXiv:1808.07784*, 2018.

[28] S. Nair and C. Finn. Hierarchical foresight: Self-supervised learning of long-horizon tasks via visual subgoal generation. *arXiv preprint arXiv:1909.05829*, 2019.

[29] K. Pertsch, O. Rybkin, F. Ebert, S. Zhou, D. Jayaraman, C. Finn, and S. Levine. Long-horizon visual planning with goal-conditioned hierarchical predictors. *Advances in Neural Information Processing Systems*, 33:17321–17333, 2020.

[30] Z. Zhang, Y. Li, O. Bastani, A. Gupta, D. Jayaraman, Y. J. Ma, and L. Weihs. Universal visual decomposer: Long-horizon manipulation made easy. *arXiv preprint arXiv:2310.08581*, 2023.

[31] C. Chi, Z. Xu, C. Pan, E. Cousineau, B. Burchfiel, S. Feng, R. Tedrake, and S. Song. Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots. In *Proceedings of Robotics: Science and Systems (RSS)*, 2024.

[32] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

[33] 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. 2022.

[34] B. Wen, W. Lian, K. Bekris, and S. Schaal. You only demonstrate once: Category-level manipulation from single visual demonstration. *arXiv preprint arXiv:2201.12716*, 2022.

[35] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[36] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[37] S. Amir, Y. Gandelsman, S. Bagon, and T. Dekel. Deep vit features as dense visual descriptors. *ECCVW What is Motion For?*, 2022.

[38] N. Di Palo and E. Johns. Dinobot: Robot manipulation via retrieval and alignment with vision foundation models. *arXiv preprint arXiv:2402.13181*, 2024.

[39] N. Di Palo and E. Johns. Keypoint action tokens enable in-context imitation learning in robotics. In *Proceedings of Robotics: Science and Systems (RSS)*, 2024.

[40] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke, et al. Scalable deep reinforcement learning for vision-based robotic manipulation. In *Conference on robot learning*, pages 651–673. PMLR, 2018.

[41] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5745–5753, 2019.

[42] B. Wen, W. Yang, J. Kautz, and S. Birchfield. Foundationpose: Unified 6d pose estimation and tracking of novel objects. *arXiv preprint arXiv:2312.08344*, 2023.

[43] B. Wen, J. Tremblay, V. Blukis, S. Tyree, T. Muller, A. Evans, D. Fox, J. Kautz, and S. Birchfield. Bundlesdf: Neural 6-dof tracking and 3d reconstruction of unknown objects. *CVPR*, 2023.

[44] J. Abou-Chakra, K. Rana, F. Dayoub, and N. Sünderhauf. Physically embodied gaussian splatting: Embedding physical priors into a visual 3d world model for robotics. In *Workshop on Neural Representations for Robotics at Conference on Robot Learning*, number 7th, 2023.

[45] J. Gao, Z. Tao, N. Jaquier, and T. Asfour. K-vil: Keypoints-based visual imitation learning. *IEEE Transactions on Robotics*, 2023.

[46] J. Gao, Z. Tao, N. Jaquier, and T. Asfour. Bi-kvil: Keypoints-based visual imitation learning of bimanual manipulation tasks. *arXiv preprint arXiv:2403.03270*, 2024.

[47] N. Heppert, M. Argus, T. Welschehold, T. Brox, and A. Valada. Ditto: Demonstration imitation by trajectory transformation. *arXiv preprint arXiv:2403.15203*, 2024.

[48] P. Wu, Y. Shentu, Z. Yi, X. Lin, and P. Abbeel. Gello: A general, low-cost, and intuitive teleoperation framework for robot manipulators, 2023.

[49] 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.