LEARNING A THOUSAND TASKS IN A DAY

Kamil Dreczkowski,* **Pietro Vitiello,*** **Vitalis Vosylius & Edward Johns**
The Robot Learning Lab at Imperial College London
{krd115, pv2017, vv19, e.johns}@imperial.ac.uk

Figure 1: Illustration of 1000 tasks taught in less than a day. The clock represents the passing of time, while each image is a frame from a real-world rollout of one of the tasks.

ABSTRACT

Humans are remarkably efficient at learning tasks from demonstrations, but today's imitation learning methods for robot manipulation often require hundreds or thousands of demonstrations per task. To bridge this gap, we discovered that decomposing reasoning into two sequential phases – object alignment and then object interaction – can enable robots to learn everyday tasks from just a single demonstration. We systematically evaluated this decomposition by comparing different design choices for each phase of reasoning, and by studying the generalisation and scaling trends with respect to today's dominant paradigm of behavioural cloning with a single-phase monolithic policy. Through 3,450 real-world policy rollouts, we found compelling conclusions that, focussing on efficient learning from few demonstrations per task, decomposition significantly outperforms learning the full trajectory in a single phase, and for each phase, reasoning via retrieval in a learned latent space outperforms behavioural cloning. Building on these insights, we then designed Multi-Task Trajectory Transfer (MT3), a novel imitation learning method based on decomposition and retrieval which is capable of learning everyday manipulation tasks from only a single demonstration each, whilst also generalising efficiently to novel objects. We found that this major leap in data efficiency ultimately enabled us to teach a robot 1000 distinct everyday tasks within just 24 hours of human demonstrator time. Videos of our experiments can be found on our anonymous website https://www.robot-learning.uk/learning-1000-tasks.

*Joint First Author Contribution

1 INTRODUCTION

Humans are remarkably efficient learners, with behaviour imitation playing a fundamental role in skill acquisition. Research shows the importance of demonstrations for efficient learning, with infants learning manipulation skills substantially faster when guided by expert demonstrations compared to unguided exploration (Somogyi et al., 2015; Fagard et al., 2016). This efficient learning through demonstration is widespread in nature, with primates learning manipulation tasks from fewer than five demonstrations (Hayes & Hayes, 1952; Horner & Whiten, 2004; Call et al., 2004; Rigamonti et al., 2005; Tennie et al., 2006) and rodents acquiring both behaviour and navigation skills from fewer than ten (Meister, 2022).

In stark contrast, robots lag far behind in learning efficiency, requiring hundreds or thousands of expert demonstrations per task (Jang et al., 2021; Jiang et al., 2022; Shafiullah et al., 2022; Brohan et al., 2023; Zitkovich et al., 2023; Zhao et al., 2023; Bharadhwaj et al., 2024; Kim et al., 2024; Black et al., 2024; Octo Model Team et al., 2024). State-of-the-art imitation learning systems using BC demonstrate this inefficiency: BC-Z required 125 hours to collect ∼26K demonstrations for 100 tasks (∼250 demonstrations per task) (Jang et al., 2021), RT-1 needed 17 months for ∼130K demonstrations across 744 tasks (∼175 demonstrations per task) (Brohan et al., 2023), MT-ACT took 2 months for 7.5K demonstrations on 38 tasks (∼200 demonstrations per task) (Bharadhwaj et al., 2024). Moreover, for very complex tasks, ALOHA Unleashed suggests the need for ∼8K demonstrations per task (Zhao et al., 2024).

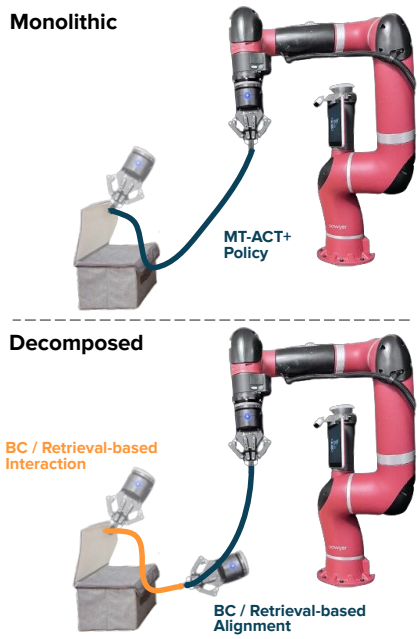

Figure 2: Decomposition of trajectories into an alignment and an interaction phase. Monolithic approaches use a single policy to handle entire trajectories, while decomposition-based approaches use two specialised policies - one to align the end-effector with target objects, and another to perform the precise manipulations. We explore both BC and retrieval-based methods for each phase of this decomposition.

While all these methods can be effective at scale, scaling them to learn thousands of tasks would require massive real-world datasets that demand enormous financial and human resources to collect. Improving learning efficiency is thus crucial to reduce the eventual data requirements for highly capable and general robotic systems. To this end, we discovered the following. These behavioural cloning methods learn reasoning with a single monolithic policy; but if reasoning is instead decomposed into two specialist, sequential phases of reasoning, we achieve an order of magnitude leap in data efficiency. Through a series of experiments exploring this decomposition and its scaling and generalisation trends, we ultimately developed a highly efficient, novel imitation learning method, Multi-Task Trajectory Transfer (MT3). To showcase its significant efficiency, we evaluated MT3 by teaching a robot one thousand distinct tasks from just a single demonstration each, in less than 24 hours of human demonstrator time (see Figure 1).

This paper presents our research that led to the emergence of MT3. First, we study the structural prior which decomposes trajectories into two sequential phases – alignment and interaction (see Figure 2). The first phase enables a robot to reason about how to align its end-effector with a target object, whilst the second phase enables a robot to reason about how to physically interact with the target object. Importantly, each phase has different motion requirements, as illustrated by a plug insertion task. While many different trajectories can successfully align the plug with the socket, the subsequent insertion demands precise control to ensure task success. We show that by using two specialised policies, one optimised for aligning with objects and the other optimised for interacting with them, we achieve significant efficiency gain compared to using a single monolithic policy to handle entire manipulation trajectories.

We then evaluate the effectiveness of this decomposition prior in learning behaviours and generalizing to novel objects, by examining – for both the alignment and interaction phases – the performance of retrieval-based methods as an alternative to standard BC methods. These methods differ in their

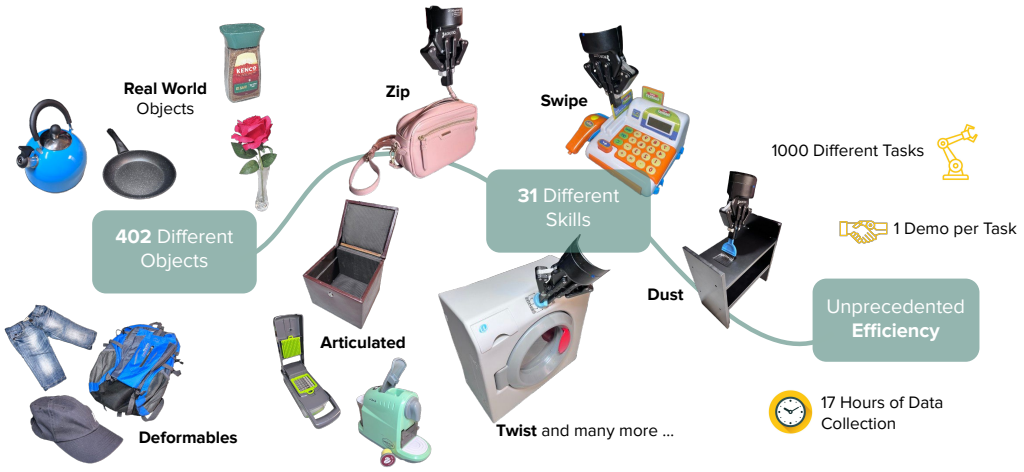

Figure 3: An overview of the 1000 evaluated tasks. This figure illustrates the diversity of our approach by showcasing examples from our collection of 402 distinct objects, highlighting some of the skills our robot has learned, and emphasizing the data efficiency achieved.

training data and inference processes. BC uses demonstrations to train a policy, and then at inference, directly predicts actions using the policy. In contrast, our retrieval-based methods do not require real-world demonstration data at training time but instead rely on it at test time as a form of guidance. As a result, we devised our own retrieval pipeline to autonomously select the best demonstration to use at inference.

Through 3,450 real-world experimental rollouts across 70 different objects, we systematically analyse the effects of the decomposition prior and retrieval-based generalisation on learning efficiency. In total, we study all four combinations of BC and retrieval-based policies when used for the two phases of a trajectory, and compare these against a standard monolithic BC method that learns entire manipulation trajectories without decomposition. By varying both the number of tasks and the number of demonstrations per task, we analyse how each method performs across different data regimes, with a focus on scenarios with limited per-task data. The results from this experiment are unambiguous: decomposing manipulation trajectories into alignment and interaction phases outperforms learning trajectories with a single monolithic policy, especially when learning from only a few demonstrations per task. Furthermore, using retrieval-based methods to align and interact with objects leads to more efficient learning than when using BC alternatives.

After establishing the effectiveness of the decomposition prior and retrieval-based methods, we conduct what is to our knowledge, the largest-scale evaluation of robot manipulation in terms of task and object diversity. Given MT3's significant learning efficiency, we found that we were able to teach a robot 1000 distinct and everyday tasks – involving interactions with over 400 objects – from a single demonstration each, in less than 24 hours (see Figure 3). This dramatically exceeds the scale of prior work, which has typically focused on learning policies to interact with fewer than 70 objects and required two orders of magnitude more demonstrations per task (Jang et al., 2021; Brohan et al., 2023; Bharadhwaj et al., 2024). Through 2,200 experimental rollouts, we shed light on MT3's performance, generalisation capabilities, and common failure modes.

2 Related Work

Trajectory Decomposition for Imitation Learning: A growing body of work demonstrates the effectiveness of decomposing manipulation trajectories into independent components. Recent approaches like Perceiver-Actor (Shridhar et al., 2022) and ChainedDiffuser (Xian et al., 2023) break down tasks into key waypoints connected by motion planning or learned controllers, achieving superior performance compared to end-to-end approaches with comparable demonstration data. Furthermore, decomposition approaches equivalent to the alignment-interaction decomposition have proven particularly effective, and have been shown to be capable of learning tasks from single demonstrations (Johns, 2021; Di Palo & Johns, 2021; Valassakis et al., 2022; Di Palo & Johns, 2024a;b). While these works focused on specific implementations, we investigate the broader effectiveness of such decompositions by exploring a wider range of learning strategies and their combinations.

Retrieval for Imitation Learning: Retrieval-based methods offer an alternative to end-to-end learning in robot manipulation. VINN (Pari et al., 2022) is an early attempt to nearest-neighbour retrieval for learning from demonstrations, storing observations and actions in a memory buffer and averaging the actions of the k most similar frames at inference. Other works have explored different retrieval strategies, such as (Du et al., 2023) which retrieves task-related data from an unlabelled buffer to train an end-to-end policy. The closest prior work, from Di Palo & Johns (2024b), also examines retrieval and decomposition for manipulation but differs in key aspects. Their study does not explore how these approaches scale with dataset size and diversity, as we do. Additionally, they compare structurally different methods, making it harder to isolate the impact of specific algorithmic design choices, as opposed to our controlled experiments. Finally, their retrieval pipeline is more limited, relying solely on RGB images rather than incorporating task descriptions and object geometries.

Scaling Up Imitation Learning for Manipulation: Recent work has demonstrated the potential of large neural networks trained on diverse robotics datasets to enable general-purpose manipulation. RT-1 (Brohan et al., 2023) showed that training on a large-scale dataset could yield a single policy capable of executing hundreds of manipulation tasks. This was extended by RT-2 (Zitkovich et al., 2023) and RoboCat (Bousmalis et al., 2024) through internet-scale vision-language pretraining to enhance generalisation. Several subsequent robot foundation models have emerged, including Octo (Octo Model Team et al., 2024), Open X-Embodiment (Vuong et al., 2023), and π_0 (Black et al., 2024), all trained on large-scale manipulation datasets. While these approaches have shown impressive capabilities, they require hundreds of demonstrations per task. In contrast, our work leverages the structural decomposition of manipulation trajectories into distinct phases of reasoning to achieve efficient learning from single demonstrations.

3 METHOD

In this work, we focus on teaching a robot multiple tasks, where each task involves a single interaction between the robot's end-effector or a grasped object, and a target object. For tasks involving grasped objects, we assume that their pose in the gripper is the same during demonstrations as testing. This formulation covers most common manipulation tasks - from grasping, to insertion, to tool usage. And while we focus on single-interaction tasks, multi-step behaviours such as pick-and-place operations can be achieved by chaining them together using existing high-level planners (see our website for videos). Our evaluation considers both seen tasks, and unseen tasks where methods must generalise to novel object instances within known categories. For clarity, we define three terms:

1. Macro skill: A verb (e.g. "unzip")
2. Micro skill: A verb plus a target object category (e.g. "unzip handbag")
3. Task: A verb plus a specific target object instance (e.g. "unzip the round pink handbag")

3.1 SYSTEM OVERVIEW

To ensure a fair comparison between using the decomposition prior (i.e. two policies - one to align and the other to interact with objects) and a single policy, we establish a consistent system architecture across all methods. The robot receives two inputs: a segmented point cloud of the target object and a language description of the task. A multi-task policy processes these inputs to generate robot actions. In terms of policy design, we compare four decomposition-based methods against a monolithic BC baseline that learns entire trajectories (see Figure 4). Below we describe the intuition behind each of these approaches.

3.2 DECOMPOSITION-BASED METHODS

The decomposition prior divides manipulation trajectories into two phases of reasoning (Figure 2):

1. **Alignment Phase**: Before interacting with an object, the robot must move its end-effector to a pose relative to the target object that is sensible for the upcoming manipulation. The specific path taken to reach this pose is not critical, provided that the robot satisfies environmental constraints during its motion. For example, in a plug insertion task, the robot can take many different paths to position the plug in front of the socket.

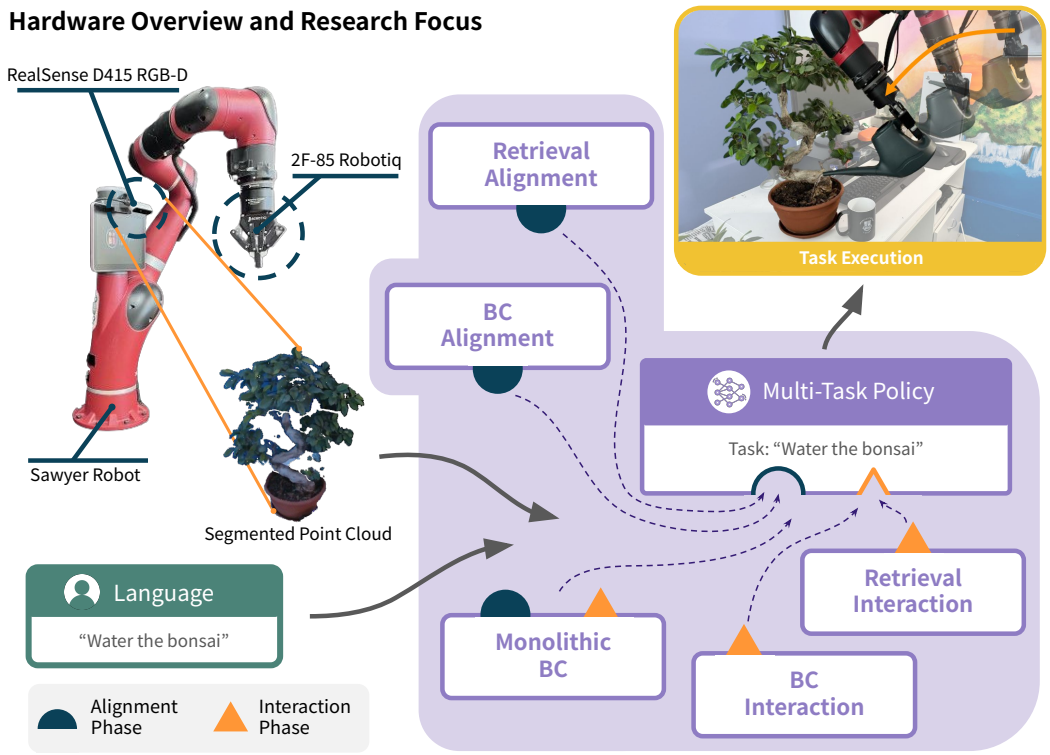

Figure 4: The puzzle pieces in the purple area are the building blocks of the five methods compared in our study. A trajectory is composed of an alignment phase (semi-sphere) followed by an interaction (triangle) phase. The monolithic policy handles both phases without making any distinction (the puzzle piece with both semi-sphere and triangle). On the other hand, decomposed methods are made by the combinations of BC or retrieval-based policies specialised in either phase. Ultimately, a policy processes a segmented point cloud and task description as input and outputs robot actions.

2. **Interaction Phase**: This phase consists in the actual manipulation and requires precise execution, as the specific trajectory is crucial for task success. For example, during the actual insertion of the plug into the socket, the motion must be carefully controlled to ensure a proper connection.

As such, all four decomposition-based methods use two policies. The first to align with objects, and the second to interact with them. Due to the different natures of the alignment and interaction phases, we show that having specialised policies for each phase can yield efficiency gains compared to using a single policy, especially when learning from few demonstrations per task. We investigate two contrasting approaches for designing the alignment and interaction policies: BC and retrieval-based methods. We explain both in the following subsections.

Behaviour Cloning Alignment and Interaction: Since BC is a prominent approach in the field of robot manipulation, we believe it can be insightful to explore what happens when this same technique is applied within the decomposition framework. BC consists in training a neural network to encode demonstrated behaviours into its weights. For our BC implementation, we chose a transformer-based backbone that employs variational inference (Shankar & Gupta, 2020; Graves, 2011), as it has demonstrated effective and efficient learning of various manipulation tasks (Bharadhwaj et al., 2024). This architecture resembles that of MT-ACT (Bharadhwaj et al., 2024) adapted to work with the expected inputs and outputs discussed in our system overview. By selectively training this architecture on demonstration data pertaining to either aligning or interacting with objects, we obtain specialised BC policies for each of the phases. More information regarding our behaviour cloning implementation can be found in Appendix A.4.

Retrieval-Based Alignment and Interaction: Instead of encoding demonstrations in network weights, retrieval-based methods store demonstrations in memory, and at test time, retrieve a single demonstration and infer actions from it (see Figure 5 for an overview and Appendix A.3 for more

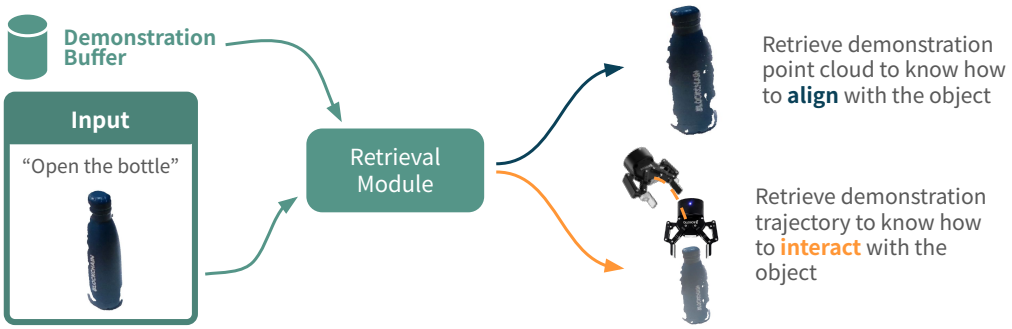

Figure 5: Retrieval-based alignment and interaction policies. Both policies use language, and a segmented point cloud to find, within a demonstration buffer, a demo for the same micro skill with the most similar object considering pose and geometry. For alignment, retrieved demonstrations show how to position relative to objects, while for interaction, they guide manipulation motions.

details). For alignment, they use pose estimation to map initial poses of demonstrations to deployment scenes (Vitiello et al., 2023) and reach these poses with motion planning. For interaction, the robot simply executes the retrieved interaction trajectory in the test scene, preserving exact motion patterns.

Resulting Decomposition-Based Methods: Combining these two approaches (BC and retrieval-based methods) across both phases (alignment and interaction), creates four distinct methods:

- **BC-BC**: **B**ehaviour **C**loning for Alignment - **B**ehaviour **C**loning for Interaction.
- **BC-Ret**: **B**ehavior **C**loning Alignment, **Ret**rieval-based Interaction.
- **Ret-BC**: **Ret**rieval-based Alignment, **B**ehaviour **C**loning Interaction.
- **Ret-Ret** (MT3): **Ret**rieval-based Alignment, **Ret**rieval-based Interaction.

Throughout this paper, we refer to Ret-Ret as Multi-Task Trajectory Transfer (MT3). MT3 extends Trajectory Transfer (Schulman et al., 2016; Vitiello et al., 2023) to multi-task learning by first using retrieval to identify the most relevant demonstration, and then using Trajectory Transfer to replicate the demonstrated task in the deployment scene.

3.3 Monolithic Behaviour Cloning Baseline

To evaluate the benefits of incorporating the decomposition prior (i.e. using two policies), we compare all four decomposition-based methods against a baseline that uses a single monolithic policy (MT-ACT+). We train this policy using BC and use the same network architecture as the BC policies used to align and interact with objects in BC-BC, BC-Ret and Ret-BC (see Appendix A.4 for details). The only difference is that instead of training it to replicate either the alignment or interaction phase of tasks, we train it to handle entire manipulation trajectories.

3.4 Different Properties of Learning Strategies

Generalisation to Unseen Object Instances: BC and retrieval-based methods represent contrasting approaches for generalising to unseen object instances. BC policies encode demonstrations in network weights, enabling interpolation between demonstrated behaviours based on geometric similarities. In contrast, retrieval-based policies do not attempt to interpolate behaviours when presented with novel instances. Instead, they identify the single closest demonstration object and treat the novel object exactly as if it were the training instance. For alignment, they position relative to the novel object as they would for the training one, while for interaction, they execute the precise demonstrated trajectory. This simplified approach is effective because optimal trajectories often maintain similar structures across object instances within a category, with task tolerance accommodating geometric variations. For example, when grasping different mugs, while sizes and handle shapes vary, the core approach and grasp motion remains consistent.

Learning from Multiple Demonstrations: Another difference between BC and retrieval-based methods is how they benefit from multiple demonstrations. BC incorporates all demonstrations during training to learn a policy that can generalise across different conditions. In contrast, retrieval-based methods do not require demonstrations during training, and instead only require them during

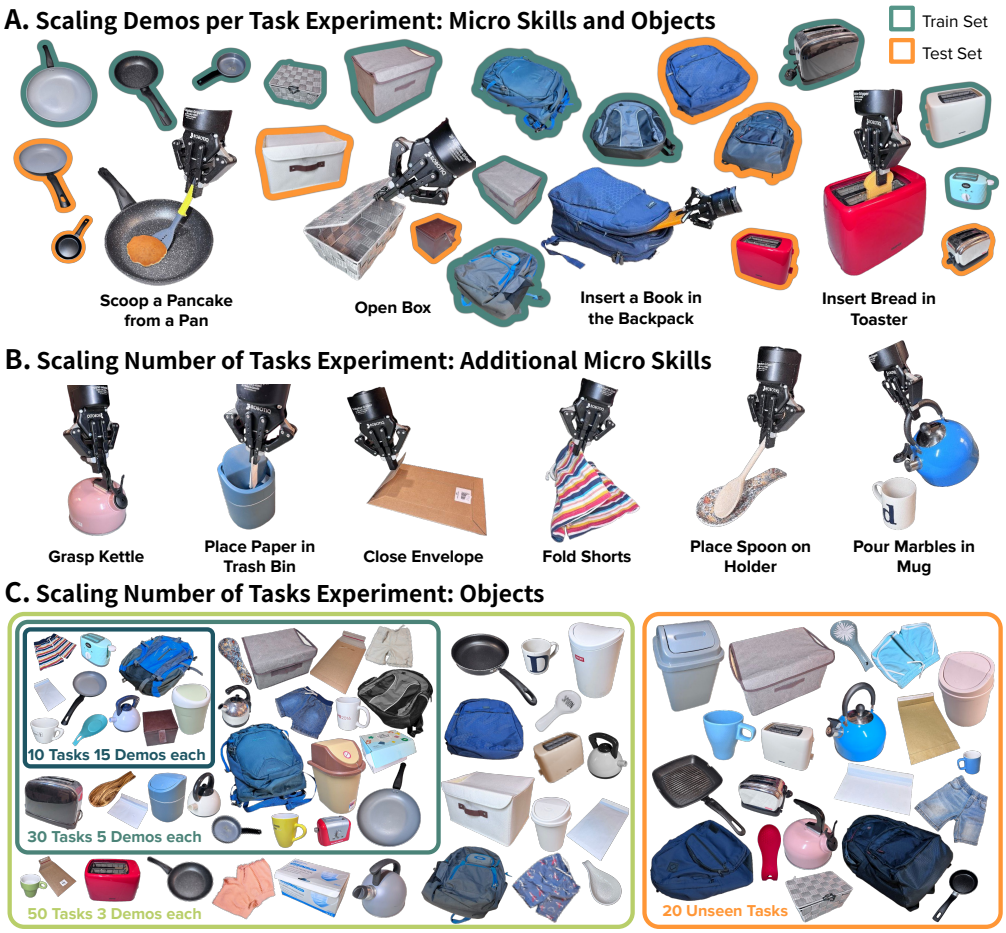

Figure 6: (A) The micro skills used to evaluate the methods' response to scaling the demonstrations per task. We also show the various seen and unseen objects used. (B) The micro skills used to evaluate the methods' response to scaling the number of tasks. These are in addition to those found in (A). (C) The objects used in the latter experiment.

the retrieval step of inference. By having access to more demonstrations, retrieval can identify a better suited demonstration for the test object instance and pose.

4 Controlled Experiments

To evaluate how each method performs across different data regimes, we design two complementary experiments that independently vary two dimensions of learning: dataset size (demonstrations per task) and dataset diversity (total number of tasks). We now provide an overview of each experiment.

Scaling Demonstrations per Task: In the first experiment, we fix the number of tasks and study how effectively each method can leverage additional demonstrations of the same tasks to improve performance. For this experiment, we select four micro skills that span diverse manipulations: articulated object manipulation, deformable object interaction, scooping, and insertion. For each micro skill, we include three seen tasks and two unseen tasks, yielding a total of 12 seen and 8 unseen tasks. The four micro skills and all tasks are shown in Figure 6.A. We evaluate the performance of all methods as we scale from a single demonstration up to 50 demonstrations per task, which has been shown to be enough for learning complex manipulation trajectories (Zhao et al., 2023).

Scaling Number of Tasks: In the second experiment, we fix the total number of demonstrations at 150 and study how performance changes as we distribute these demonstrations across an increasing number of tasks. We do this to investigate whether performance degrades when we have fewer demonstrations per task, or if some methods can benefit from exposure to more object instances

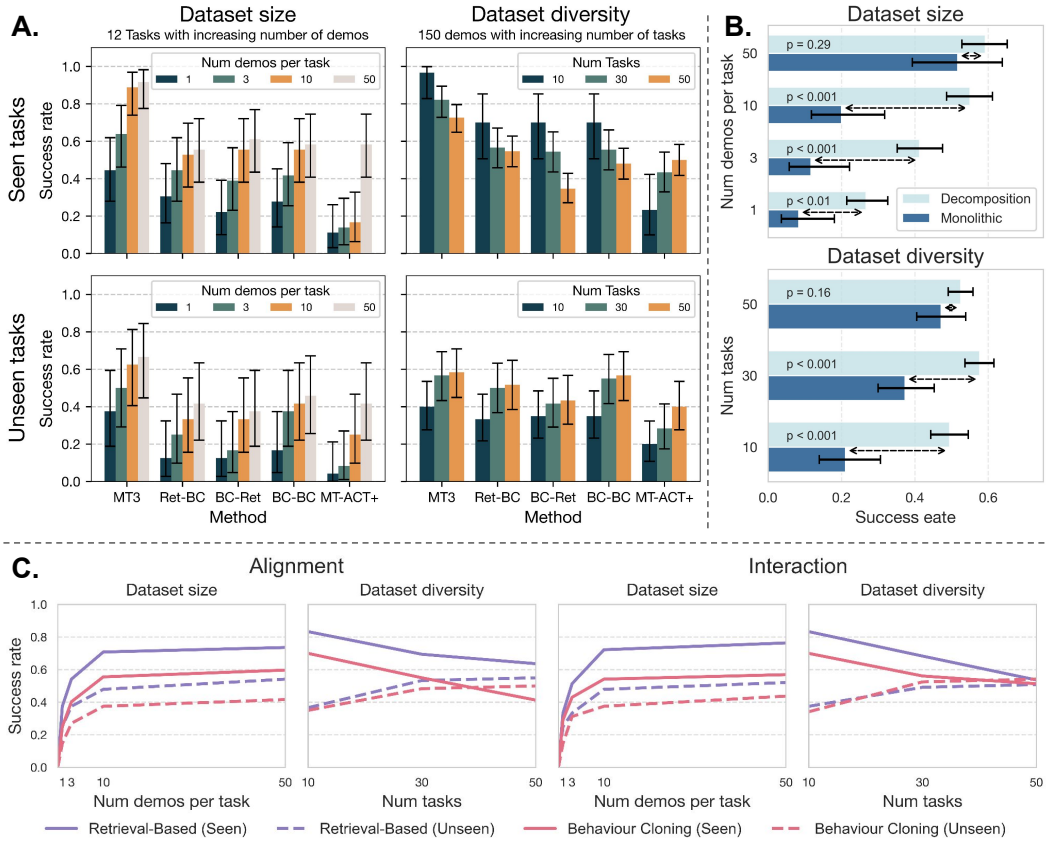

Figure 7: Analysis of dataset size and diversity effects on task performance. (A) Performance comparison across all considered methods, with error bars showing Wilson confidence intervals. (B) Comparison between decomposition-based approaches (aggregated results from Ret-Ret (MT3), Ret-BC, BC-Ret, and BC-BC) and monolithic learning (MT-ACT+), averaged across seen and unseen tasks, with error bars showing Wilson confidence intervals, and statistical significance assessed using the two-proportion Z-test. (C) Analysis of strategies to align and interact with objects: alignment plots compare behaviour cloning (BC-BC, BC-Ret) versus retrieval-based methods (Ret-BC, Ret-Ret (MT3)) to align with objects, while interaction plots compare behaviour cloning (BC-BC, Ret-BC) versus retrieval-based methods (BC-Ret, Ret-Ret (MT3)) to interact with objects. Success rates are shown as a function of dataset size (number of demonstrations per task) and diversity (number of tasks).

even with fewer demonstrations per task. For this experiment, we select 10 different micro skills, which we show in Figure 6.A and 6.B. We start with 10 tasks with 15 demonstrations each, then increase to 30 tasks with 5 demos each, and finally 50 tasks with 3 demos each. For each diversity regime, we maintain consistent evaluation across 2 unseen tasks per micro skill (20 unseen tasks total). All objects used for this experiment are shown in Figure 6.C.

Evaluation Procedure: For both experiments we conduct three evaluations per task and average results across all micro skills (see Appendix A.5 for more details). All results from these experiments are shown in Figure 7.

4.1 Individual Method Performance

Figure 7.A shows the results from our dataset size and diversity experiments across all evaluated methods. These results reveal a clear performance hierarchy among the evaluated methods. MT3, the fully retrieval-based method, consistently demonstrates superior performance across all considered data regimes. This is particularly evident when noticing that for both seen and unseen tasks, MT3 with just three demonstrations per task outperforms all other methods even when they are provided with fifty demonstrations per task. The strong performance of MT3 on unseen tasks demon-

strates that while being simple, retrieval is a viable approach for tackling generalisation to unseen object instances. Decomposition also shown its benefits, with the remaining methods relying on it (Ret-BC, BC-Ret and BC-BC) generally outperforming the monolithic baseline MT-ACT+.

4.2 Superior Efficiency Through Decomposition

Next, we investigate whether decomposing manipulation trajectories into alignment and interaction phases provides benefits on average over learning complete trajectories end-to-end. To isolate the impact of the decomposition prior from specific design choices, Figure 7.B averages the results shown in Figure 7.A across all decomposition-based methods (MT3, Ret-BC, BC-Ret and BC-BC). The average decomposition performance is then compared against MT-ACT+'s results (monolithic), with success rates for both the average decomposition performance and MT-ACT+ aggregated across seen and unseen tasks.

Distinct Scaling Patterns: The dataset size results (top Figure 7.B) demonstrate fundamentally different learning dynamics between the approaches. Explicitly leveraging the natural separation of manipulation tasks into alignment and interaction phases shows rapid improvement in the critical 1-10 demonstrations per task range, with the average decomposition performance with just one demonstration per task surpassing that of MT-ACT+ when learning with up to 10 demonstrations per task. However, improvements in performance seem to approach saturation for decomposition-based methods when given 50 demonstrations per task. On the other hand, for the monolithic baseline, initial progress is slow, but performance increases substantially when increasing from 10 to 50 demonstrations per task, narrowing the gap in performance with the decomposition-based methods. This highlights how the monolithic strategy requires substantially more data to learn the task structure that decomposition-based methods leverage by construction.

Response to Task Diversity: When maintaining a fixed budget of 150 demonstrations but varying their distribution across tasks, we uncover striking differences. As seen from the two rightmost plots in Figure 7.A, decomposition-based methods achieve peak performance when focussed on fewer tasks, with seen-task performance declining as demonstrations spread thinner across more tasks. However, unseen task performance still improves because having demonstrations distributed across more diverse objects increases the likelihood that the demonstration data includes an object with similar geometry to the test object. Conversely, MT-ACT+ shows improvement in both seen and unseen task performance with increased task diversity. This suggests that learning complete trajectories might facilitate BC's ability to find patterns across manipulations of different object instances. However, it is worth noting that even though the performance trend is better for the monolithic approach, its absolute performance is still lower than that of decomposition-based methods.

Strength of Decomposition: Overall, the results are clear: on average, decomposing manipulation trajectories into alignment and interaction phases is beneficial across all tested experimental conditions. Moreover, the benefit of trajectory decomposition holds even when the underlying method remains unchanged - as shown in Figure 7.A, BC-BC substantially outperforms MT-ACT+ despite both fundamentally relying on the same BC implementation. This advantage stems from the distinct properties of alignment and interaction phases, best leveraged by specialised policies.

4.3 Surprising Effectiveness of Retrieval

In this section, we are interested in how BC and retrieval compare when used to align and interact with objects, with Figure 7.C showing the comparison of these two approaches within the decomposition framework. To generate these graphs, we take all the results for the decomposition-based methods (MT3, Ret-BC, BC-Ret and BC-BC) shown in Figure 7.A. Then for each phase (alignment and interaction), we aggregate the task success rates across methods that share the same strategy. For example, combining MT3 (Ret-Ret) and Ret-BC to evaluate retrieval-based alignment, while combining BC-Ret and BC-BC to evaluate alignment using BC.

Performances and Trends: Our analysis shows that retrieval-based strategies outperform BC across all experimental conditions when used to both align and interact with objects. All the while, the trends in performance across all data regimes seem very similar between retrieval and BC approaches, suggesting that both can benefit from additional data or diversity in a similar manner.

Interacting with Novel Object Instances: Surprisingly, even though retrieval-based interaction methods simply replicate a demonstration's motion, they work well for generalising to unseen objects. The effectiveness of this approach stems from two key insights. First, optimal interaction trajectories often remain similar across different instances of the same object category, even when their geometry varies substantially. Second, many manipulation tasks exhibit natural tolerance to variations in object geometry. Yet, it is worth noting that these insights also benefit BC approaches.

5 Scaling to a Thousand Tasks

While numerous studies have explored scaling up monolithic BC across both tasks and demonstrations using large numbers of demonstrations per task (Bharadhwaj et al., 2024; Brohan et al., 2023; Zitkovich et al., 2023; Zhao et al., 2024), far less attention has been paid to the challenge of scaling any BC alternative, especially while relying only on a few demonstrations per task. Our controlled experiments demonstrate MT3's superior performance without requiring large demonstration datasets, yet a crucial question remains: could this efficiency scale to learning a truly diverse range of real-world manipulation tasks?

Figure 8: Example test scenes from MT3 evaluation. Each scene contains 5-20 distractor objects with varied backgrounds and randomised lighting conditions.

To answer this, we conducted an unprecedented robotic manipulation study, teaching a robot 1000 distinct manipulation tasks in under 24 hours using just a single demonstration per task. This represents the first work to demonstrate learning manipulation skills at this scale without relying on large demonstration datasets, dramatically surpassing previous studies which typically focused on tens or hundreds of tasks while requiring many more demonstrations per task. The focus on diversity is further underscored when considering that the considered tasks fall under 31 different macro skills and make use of 402 different objects.

Setting an Ambitious Challenge: The scale of this experiment marks a significant departure from prior work in robot learning. Recent approaches have typically focused on learning less than 750 tasks from extensive demonstration datasets collected over long periods (Jang et al., 2021; Brohan et al., 2023; Bharadhwaj et al., 2024). In contrast, our decomposition prior enabled collection of demonstrations for all 1000 tasks within 24 hours using a single robot, while maintaining comparable task complexity and achieving significantly higher task diversity.

This efficiency gain allowed us to explore a far broader range of manipulations than previously possible. Our tasks span 31 distinct macro skills (e.g., "pour", "insert", "fold") and 534 micro skills (e.g. "pour wine from wine bottle into wine glass", "pour milk from carton into bowl", "insert plate into plate rack" "insert plug into socket", "fold towel", "fold t-shirt"), representing most common household manipulation scenarios. This diversity dwarfs both our earlier controlled experiment (10 micro skills) and prior work, which typically focused on 9-12 macro skills within a more constrained set of objects, 12 to 70 (Jang et al., 2021; Brohan et al., 2023; Bharadhwaj et al., 2024), compared to our 402 different instances. To rigorously evaluate generalisation, we further tested on 100 additional unseen tasks spanning the same set of macro skills.

This efficiency gain allowed us to explore a far broader range of manipulations than previously possible. Our tasks span 31 distinct macro skills (e.g., "pour", "insert", "fold") and 534 micro skills (e.g. "pour wine from wine bottle into wine glass", "pour milk from carton into bowl", "insert plate into plate rack" "insert plug into socket", "fold towel", "fold t-shirt"), representing most common household manipulation scenarios. This diversity dwarfs both our earlier controlled experiment (10

micro skills) and prior work, which typically focused on 9-12 macro skills within a more constrained set of objects, 12 to 70 (Jang et al., 2021; Brohan et al., 2023; Bharadhwaj et al., 2024), compared to our 402 different instances. To rigorously evaluate generalisation, we further tested on 100 additional unseen tasks spanning the same set of macro skills. Furthermore, we deliberately designed the experiment to stress-test MT3's capabilities across multiple dimensions (see Figure 8 for example test scenes and Appendix A.6 for further detail on our experimental design). Hereafter we discuss the results of our evaluation of MT3, which consisted of 2,200 total rollouts (Figure 12 in Appendix A.6), with two trials per task for both seen and unseen categories.

Task Complexities: Understanding Performance Patterns: MT3 achieved a 78.25% average success rate on seen tasks and 65.66% on unseen tasks (see Appendix A.6 for detailed failure case analysis) - strong results that gain additional significance given that only a single demonstration was provided per seen task, together with the unprecedented task diversity and challenging real-world conditions.

Task performance across different macro skills (Figure 9) reveals that the success rates strongly correlate with precision requirements, as expected. Tasks with high tolerance to imperfections in execution, such as stacking and dusting, achieved success rates above 80-90%. These macro skills permit small deviation in approach angles and contact positions while rarely affecting task completion. In contrast, tasks demanding precise execution like insertions and hanging objects achieved lower success rates, reflecting their lower tolerance for execution errors. These results illustrate that MT3 is a viable approach for learning a very large number of diverse tasks from minimal data. Moreover, generalisation performance on unseen tasks highlights how a purely retrieval-based approach can still effectively bridge the gap between demonstrated and novel instances, while relying on explicit geometric reasoning rather than data-intensive learning.

Our evaluation of MT3 has also generated a rich dataset of robot execution rollouts, which we open source. These rollouts, collected under challenging real-world conditions with diverse objects and backgrounds represent a valuable resource for future developments.

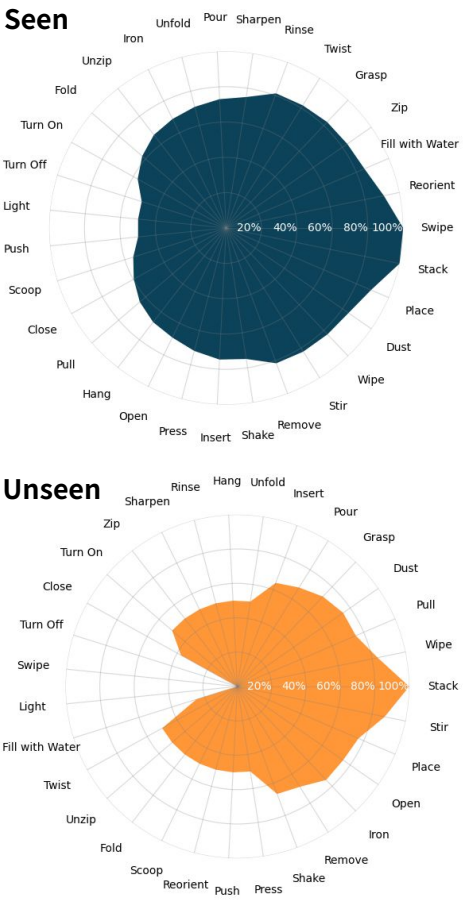

Figure 9: Success rates of MT3 across different macro skills for seen (top) and unseen (bottom) tasks.

6 Conclusions

In this paper, we demonstrate that decomposing manipulation trajectories into distinct alignment and interaction phases enables significant improvements in learning efficiency compared to a standard monolithic BC approach. While state-of-the-art BC systems typically require hundreds of demonstrations per task collected over months, our large-scale evaluation involving 3,450 real-world rollouts shows that decomposition-based approaches achieve effective learning from as little as a single demonstration per task. Furthermore, our systematic evaluation reveals that retrieval-based methods are viable and competitive alternatives to BC for both aligning and interacting with objects when the data per task is scarce.

These findings led us to the development of MT3, our highly-efficient fully retrieval-based method. In order to discover how MT3's effectiveness would scale, we evaluated at an unprecedented scale of task diversity. We have taught our system 1000 distinct tasks, manipulating over 400 objects, using just a single demonstration per task, in under 24 hours. This large-scale evaluation provided valuable insights into MT3's performance across different types of manipulation skills and revealed specific failure modes, pointing to clear paths for future improvements.

REFERENCES

Homanga Bharadhwaj, Jay Vakil, Mohit Sharma, et al. RoboAgent: Generalization and Efficiency in Robot Manipulation via Semantic Augmentations and Action Chunking. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4788–4795, 2024. doi: 10.1109/ICRA57147.2024.10611293.

Kevin Black, Noah Brown, Danny Driess, et al. π_0: A vision-language-action flow model for general robot control, 2024. URL https://arxiv.org/abs/2410.24164.

Konstantinos Bousmalis, Giulia Vezzani, Dushyant Rao, et al. Robocat: A self-improving generalist agent for robotic manipulation. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=vsCpILiWHu.

Anthony Brohan, Noah Brown, Justice Carbajal, et al. RT-1: robotics transformer for real-world control at scale. In Kostas E. Bekris, Kris Hauser, Sylvia L. Herbert, and Jingjin Yu (eds.), *Robotics: Science and Systems XIX, Daegu, Republic of Korea, July 10-14, 2023*, 2023. doi: 10.15607/RSS.2023.XIX.025. URL https://doi.org/10.15607/RSS.2023.XIX.025.

Josep Call, Malinda Carpenter, and Michael Tomasello. Copying results and copying actions in the process of social learning: chimpanzees (pan troglodytes) and human children (homo sapiens). *Animal Cognition*, 8(3):151–163, October 2004.

Ho Kei Cheng and Alexander G. Schwing. XMem: Long-term video object segmentation with an atkinson-shiffrin memory model. In *ECCV*, 2022.

Norman Di Palo and Edward Johns. Learning Multi-Stage Tasks with One Demonstration via Self-Replay. In *Conference on Robot Learning (CoRL)*, 2021.

Norman Di Palo and Edward Johns. DINOBot: Robot Manipulation via Retrieval and Alignment with Vision Foundation Models. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2024a.

Norman Di Palo and Edward Johns. On the effectiveness of retrieval, alignment, and replay in manipulation. *RA-Letters*, 2024b.

Maximilian Du, Suraj Nair, Dorsa Sadigh, and Chelsea Finn. Behavior Retrieval: Few-Shot Imitation Learning by Querying Unlabeled Datasets. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023. doi: 10.15607/RSS.2023.XIX.011.

Jacqueline Fagard, Lauriane Rat-Fischer, Rana Esseily, et al. What does it take for an infant to learn how to use a tool by observation? *Frontiers in Psychology*, 7:267, March 2016.

Alex Graves. Practical variational inference for neural networks. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K.Q. Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc., 2011. URL https://proceedings.neurips.cc/paper_files/paper/2011/file/7eb3c8be3d411e8ebfab08eba5f49632-Paper.pdf.

Keith J. Hayes and Catherine Hayes. Imitation in a home-raised chimpanzee. *Journal of Comparative and Physiological Psychology*, 45(5):450–459, 1952. ISSN 0021-9940(Print). doi: 10.1037/h0053609. URL https://doi.org/10.1037/h0053609.

Victoria Horner and andrew Whiten. Causal knowledge and imitation/emulation switching in chimpanzees (pan troglodytes) and children (homo sapiens). *Animal Cognition*, 8(3):164–181, November 2004.

Eric Jang, Alex Irpan, Mohi Khansari, et al. BC-z: Zero-shot task generalization with robotic imitation learning. In *5th Annual Conference on Robot Learning*, 2021. URL https://openreview.net/forum?id=8kbp23tSGYv.

Yunfan Jiang, Agrim Gupta, Zichen Zhang, et al. Vima: General robot manipulation with multimodal prompts. In *NeurIPS 2022 Foundation Models for Decision Making Workshop*, 2022.

Edward Johns. Coarse-to-Fine Imitation Learning: Robot Manipulation from a Single Demonstration. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.

Alex Kendall and Roberto Cipolla. Geometric loss functions for camera pose regression with deep learning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6555–6564, 2017. doi: 10.1109/CVPR.2017.694.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

Minghua Liu, Xuanlin Li, Zhan Ling, et al. Frame mining: a free lunch for learning robotic manipulation from 3d point clouds. In *6th Annual Conference on Robot Learning*, 2022. URL https://openreview.net/forum?id=d-JYso87y6s.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 38–55, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-72970-6.

Markus Meister. Learning, fast and slow. *Current Opinion in Neurobiology*, 75:102555, 2022. ISSN 0959-4388. doi: https://doi.org/10.1016/j.conb.2022.102555. URL https://www.sciencedirect.com/science/article/pii/S0959438822000496.

Octo Model Team, Dibya Ghosh, Homer Walke, et al. Octo: An open-source generalist robot policy. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, 2024.

Georgios Papagiannis, Kamil Dreczkowski, Vitalis Vosylius, et al. Adapting skills to novel grasps: A self-supervised approach. In *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, 2024.

Jyothish Pari, Nur Muhammad (Mahi) Shafiullah, Sridhar Pandian Arunachalam, and Lerrel Pinto. The Surprising Effectiveness of Representation Learning for Visual Imitation. In *Proceedings of Robotics: Science and Systems*, New York City, NY, USA, June 2022. doi: 10.15607/RSS.2022.XVIII.010.

Ethan Perez, Florian Strub, Harm de Vries, et al. FiLM: Visual reasoning with a general conditioning layer. In *AAAI*, 2018.

Charles Ruizhongtai Qi, Li Yi, Hao Su, et al. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/d8bf84be3800d12f74d8b05e9b89836f-Paper.pdf.

Alec Radford, Jong Wook Kim, Chris Hallacy, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021. URL https://api.semanticscholar.org/CorpusID:231591445.

Marco M Rigamonti, Deborah M Custance, Emanuela Prato Previde, et al. Testing for localized stimulus enhancement and object movement reenactment in pig-tailed macaques (macaca nemestrina) and young children (homo sapiens). *Journal of Comparative Psychology*, 119(3):257–272, August 2005.

John Schulman, Jonathan Ho, Cameron Lee, et al. *Learning from Demonstrations Through the Use of Non-rigid Registration*, pp. 339–354. Springer International Publishing, Robotics Research: The 16th International Symposium ISRR, Cham, 2016. ISBN 978-3-319-28872-7. doi: 10.1007/978-3-319-28872-7_20. URL https://doi.org/10.1007/978-3-319-28872-7_20.

A. Segal, D. Haehnel, and S. Thrun. Generalized-ICP. In *Proceedings of Robotics: Science and Systems*, Seattle, USA, June 2009. doi: 10.15607/RSS.2009.V.021.

Nur Muhammad Shafiullah, Zichen Cui, Ariuntuya Arty Altanzaya, et al. Behavior transformers: Cloning k modes with one stone. *Advances in neural information processing systems*, 35:22955–22968, 2022.

Tanmay Shankar and Abhinav Gupta. Learning robot skills with temporal variational inference, 2020. URL https://arxiv.org/abs/2006.16232.

Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Perceiver-actor: A multi-task transformer for robotic manipulation. In *6th Annual Conference on Robot Learning*, 2022. URL https://openreview.net/forum?id=PS_eCS_WCvD.

Eszter Somogyi, Cecilia Ara, Eugenia Gianni, et al. The roles of observation and manipulation in learning to use a tool. *Cognitive Development*, 35:186–200, 2015. ISSN 0885-2014. doi: https://doi.org/10.1016/j.cogdev.2015.06.002. URL https://www.sciencedirect.com/science/article/pii/S0885201415000209.

Claudio Tennie, Josep Call, and Michael Tomasello. Push or pull: Imitation vs. emulation in great apes and human children. *Ethology*, 112(12):1159–1169, 2006. doi: https://doi.org/10.1111/j.1439-0310.2006.01269.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1439-0310.2006.01269.x.

Eugene Valassakis, Georgios Papagiannis, Norman Di Palo, et al. Demonstrate Once, Imitate Immediately (DOME): Learning Visual Servoing for One-Shot Imitation Learning. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022.

Pietro Vitiello, Kamil Dreczkowski, and Edward Johns. One-Shot Imitation Learning: A Pose Estimation Perspective. In *Conference on Robot Learning*, 2023.

Quan Vuong, Sergey Levine, Homer Rich Walke, et al. Open X-Embodiment: Robotic learning datasets and RT-x models. In *Towards Generalist Robots: Learning Paradigms for Scalable Skill Acquisition @ CoRL2023*, 2023. URL https://openreview.net/forum?id=zraBtFgxT0.

Zhou Xian, Nikolaos Gkanatsios, Theophile Gervet, et al. Chaineddiffuser: Unifying trajectory diffusion and keypose prediction for robotic manipulation. In Jie Tan, Marc Toussaint, and Kourosh Darvish (eds.), *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 2323–2339. PMLR, 06–09 Nov 2023. URL https://proceedings.mlr.press/v229/xian23a.html.

Tony Z. Zhao, Vikash Kumar, Sergey Levine, et al. Learning fine-grained bimanual manipulation with low-cost hardware. In Kostas E. Bekris, Kris Hauser, Sylvia L. Herbert, and Jingjin Yu (eds.), *Robotics: Science and Systems XIX, Daegu, Republic of Korea, July 10-14, 2023*, 2023. doi: 10.15607/RSS.2023.XIX.016. URL https://doi.org/10.15607/RSS.2023.XIX.016.

Tony Z. Zhao, Jonathan Tompson, Danny Driess, et al. ALOHA unleashed: A simple recipe for robot dexterity. In *8th Annual Conference on Robot Learning*, 2024. URL https://openreview.net/forum?id=gvdXE7ikHI.

Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3D: A modern library for 3D data processing. *arXiv:1801.09847*, 2018.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, et al. RT-2: Vision-language-action models transfer web knowledge to robotic control. In *7th Annual Conference on Robot Learning*, 2023. URL https://openreview.net/forum?id=XMQgwiJ7KSX.

A APPENDIX

A.1 HARDWARE OVERVIEW

Our experimental hardware consists of a Sawyer robot equipped with a 2F-85 Robotiq gripper and is shown in Figure 4. For perception, we use a single RealSense D415 RGB-D camera mounted on the robot's head, providing sufficient visual information for manipulation tasks while minimising hardware costs.

A.2 DEMONSTRATION DATA COLLECTION AND PROCESSING

In this section, we explain how we represent, collect and process demonstrations for all methods.

Demonstration Representation: We denote a demonstration $\tau = \{o_i, e_i\}_{i=1}^{N}$ as a sequence of observations o and end-effector states e recorder at 30 Hz, where i indexes time-steps and N is the sequence length. Each observation o_i is an RGB-D image from a calibrated head-mounted camera. The corresponding end-effector state e_i includes the 6D pose of the end-effector frame E in the robot's base frame R, $\boldsymbol{T}_{RE} \in SE(3)$, and the binary gripper state that indicates if the gripper is opened or closed. Each demonstration is paired with a language description l to differentiate between tasks. This creates a dataset D of M demonstrations and their corresponding descriptions: $D = \{\tau_j, l_j\}_{j=1}^{M}$.

Demonstration Data Collection: During data collection, we record only the interaction phase of each task. This is because, in the alignment phase, the critical factor is the final pose of the end-effector relative to the object, rather than the specific trajectory taken to reach it. In contrast, the interaction phase defines how the object should be manipulated, making it rich in task-relevant information. In practice, we start recording demonstrations only once the end-effector is positioned close to the target object (the exact pose does not matter as long as the end-effector is in the proximity of the target object).

This formulation offers two key advantages: (1) Since only the final pose matters for alignment, we can generate synthetic alignment trajectories when needed, as detailed in Appendix A.4. (2) The decomposition of demonstrations into two phases becomes straightforward — interaction trajectories are collected from real-world demonstrations, while alignment trajectories are synthetically generated.

Demonstration Data Processing: For each of our methods, we segment all RGB-D images and convert them to target object point clouds. We segment the images to enhance efficiency and robustness against background changes and distractors. To this end, for the initial RGB image of each demonstration, we use Grounding DINO (Liu et al., 2025) to segment the target object using the target object name extracted from the task description l. For simplicity, we extract this name using template-based natural language parsing, though more sophisticated approaches using large language models could be employed. For each subsequent frame of each demonstration, we propagate the target object segmentation using XMem (Cheng & Schwing, 2022), which handles partial and full occlusions.

RGB-D images and segmentation masks are then processed into target object point clouds using known camera parameters and serve as our policy inputs. For retrieval-based methods, we express these point clouds in the robot frame as this is required by the pose estimator used by the retrieval-based method for alignment (see Section A.3). For training BC policies, we transform the point clouds into the end-effector frame to improve learning efficiency and spatial generalisation (Liu et al., 2022).

A.3 RETRIEVAL-BASED ALIGNMENT AND INTERACTION

We designed two different retrieval-based policies, one to align with objects and the other to interact with them. At test time, both policies require a single demonstration of the desired task to infer actions from. As such, they both rely on a retrieval system to autonomously identify the demonstration that best matches the test scenario by finding one which performs the same manipulation on a similar object.

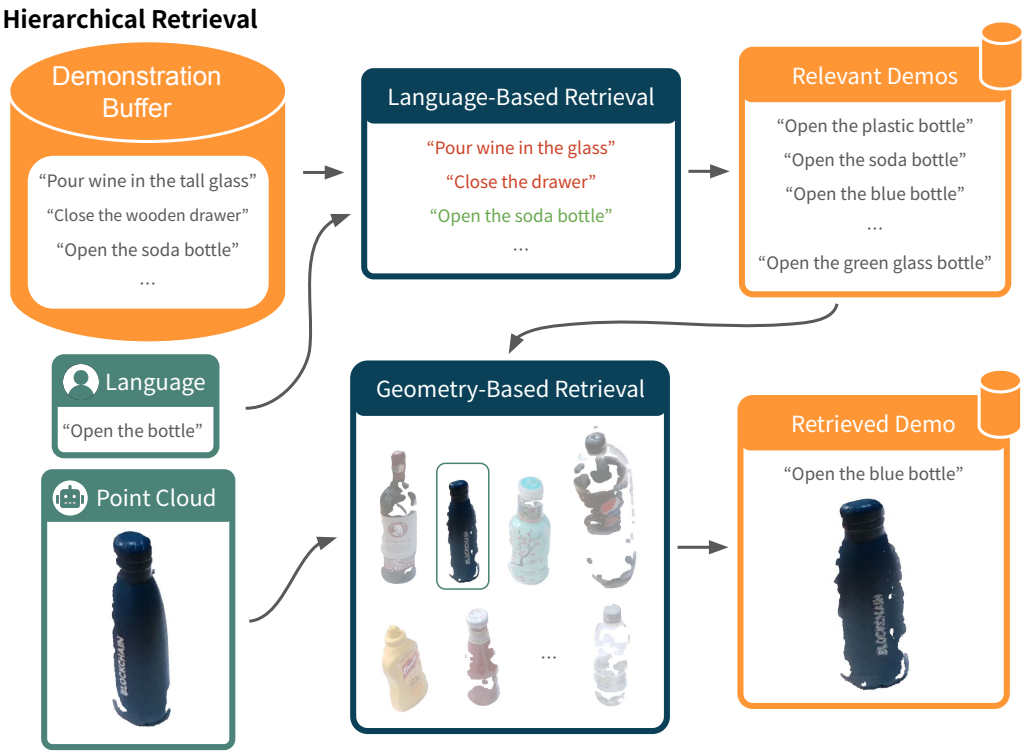

Figure 10: Our hierarchical retrieval pipeline consists of two stages. First, language-based retrieval identifies all demonstrations associated with the specific micro-skill mentioned in the task description, drawing from the entire dataset of available demonstrations. Second, geometry-based retrieval refines this selection by finding the closest demonstration in terms of object shape and pose.

After retrieving a demonstration, the policies differ in how they use it. The alignment policy uses pose estimation to map the initial end-effector pose from the demonstration to the test scene, followed by motion planning to reach this pose. The interaction policy directly replays the demonstrated trajectory in the end-effector frame, preserving the exact motion patterns. Below, we detail each of these components.

Hierarchical Retrieval: Our retrieval system uses a two-stage approach that is illustrated in Figure 10. In the first stage, we use the micro skill name inferred from the task description (e.g., "open bottle") to find all demonstrations for that same micro skill. For simplicity, we extract this micro skill name from the task description l using a template matching approach, though more sophisticated approaches using large language models could be used.

In the second stage, we identify the demonstration with the most similar object to the test object in terms of geometry and object pose. We rely on geometry as we believe that objects that have similar shapes and sizes will also require similar interactions. Additionally, selectivity in object pose allows us to retrieve demonstrations that are as close as possible to the test scene, narrowing the covariate shift. To capture geometry and pose similarity for the purpose of retrieval, we use a point cloud encoder based on the PointNet++ (Qi et al., 2017) architecture to generate object embeddings. The encoder was trained as part of an auto-encoder, which compresses a point cloud into an embedding that is then decoded to predict an occupancy grid trained with binary cross-entropy using the object-centric dataset generated as part of prior work (Vitiello et al., 2023). The demonstration with the highest cosine similarity between its object embedding and the test object embedding is selected.

Figure 11 shows a t-SNE plot of object embeddings from one of our controlled experiments (learning 12 tasks with 50 demonstrations per task - see Section 4). This plot reveals clustering by object category (e.g., backpacks, toasters) with subclusters corresponding to different object instances, demonstrating the encoder's ability to capture both broad category-level features and fine-grained

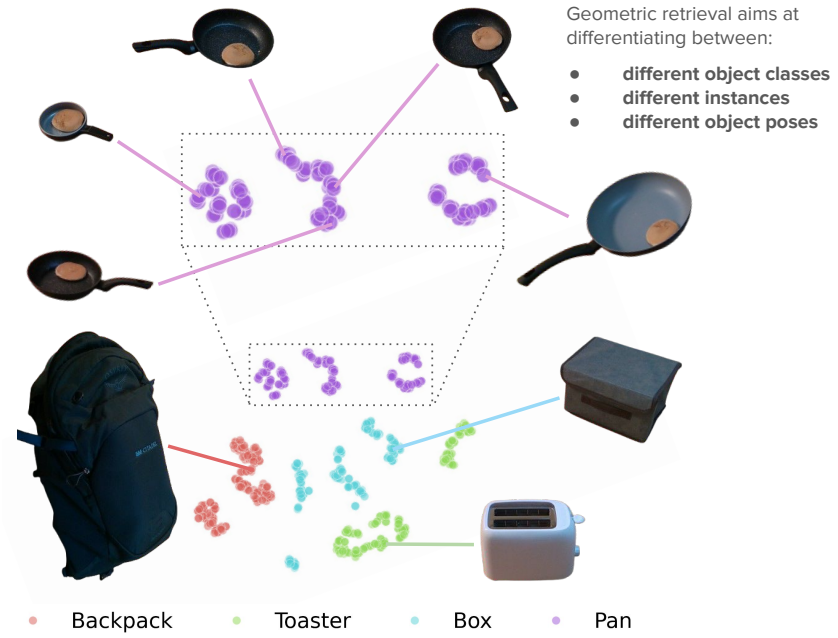

Figure 11: t-SNE visualisation of geometry encodings from the dataset size experiment with 50 demonstrations per object (see Section 4), showing clustering by object category (backpack, toaster, box, and pan). Each category exhibits subclusters corresponding to different object instances, with similar geometries (box, toaster) positioned closer in the embedding space. Within subclusters, points from similar object poses have closer embeddings.

geometric differences. This hierarchical organisation of the embedding space is crucial for generalisation in two ways. First, objects with similar global geometry are mapped to nearby regions in the embedding space, making it more likely to retrieve demonstrations of interactions with objects that share similar manipulation requirements. Second, the clustering of similar poses within each object's subcluster helps match novel object poses to demonstrations where the object was in a similar pose, enabling more relevant demonstration selection.

Retrieval-Based Alignment: At inference, the retrieval-based alignment policy receives the target object point cloud and a demonstration of the desired task, and its goal is to align the end-effector and the target object in the same way as shown at the beginning of the demonstration. To this end, the policy first uses geometric reasoning to infer the required end-effector pose for the test scene and then reaches this pose through motion planning.

In this work, we calculate the end-effector pose for the test scene that aligns the end-effector and target object in the same way as shown at the beginning of the demonstration using Trajectory Transfer (Vitiello et al., 2023). The intuition behind Trajectory Transfer is that given the relative target object pose between the demonstration and test scene $\boldsymbol{T}_\delta$, we can map the end-effector pose at the beginning of the demonstration to the test scene using

$$\boldsymbol{T}_{WE}^{Test} = \boldsymbol{T}_\delta \boldsymbol{T}_{WE}^{Demo}$$

where $\boldsymbol{T}_{WE}^{Test}$ and $\boldsymbol{T}_{WE}^{Demo}$ are the end-effector poses for the test and demonstration scenes respectively that correspond to the same end-effector to target object pose. We estimate $\boldsymbol{T}_\delta$ by refining the output of the regression method proposed by Vitiello et al. (2023) using the Open3D (Zhou et al., 2018) implementation of Generalised ICP (Segal et al., 2009).

Retrieval-Based Interaction: Similar to the retrieval-based alignment policy, at inference, the interaction policy receives a demonstration of the desired task. The demonstrated trajectory is then replicated in the test scene by executing the demonstrated end-effector velocities expressed in the end effector frame, preserving the exact demonstrated motion patterns.

A.4 BEHAVIOURAL CLONING IMPLEMENTATION

We use the same network architecture and loss function to learn to align and interact with objects, and to learn the single policy for the MT-ACT+ baseline. The only difference between these applications is the training data they rely on. Below we describe our chosen backbone architecture, the loss function used, and the data all these policies have been specifically trained on.

Network Architecture and Design Choices: For all three applications, we need the policy architecture to address three requirements. First, it must process point cloud and language inputs for a fair comparison with our retrieval-based components. Second, it must effectively handle multi-task learning to enable a comparison with the retrieval-based policies across diverse tasks. Third, it needs to capture the multi-modal nature of manipulation demonstrations, where multiple trajectories may be valid for completing the same task (or phase of the task).

To handle point cloud inputs, we employ a PointNet++ (Qi et al., 2017) encoder which clusters the point clouds and computes an embedding per cluster. For multi-task learning, we condition these embeddings on the task description using FiLM (Perez et al., 2018). This modulation takes a CLIP (Radford et al., 2021) embedding of the task description l and uses it to adapt the point cloud features for the specific task at hand. To address the multi-modal nature of demonstrations, we employ variational inference, which enables the policy to model the multi-modal distribution of valid actions. While diffusion models offer an alternative approach, variational inference provides a more computationally efficient solution, while still being suitable for modelling multi-modal distributions.

This combination of design choices results in an architecture that closely resembles the MT-ACT architecture proposed by Bharadhwaj et al. (2024), with modifications to handle point cloud inputs. Additional differences from MT-ACT include incorporating action history as input to help infer task progress, removing proprioception from the input which our preliminary experiments showed improved spatial generalisation, and adding a terminal action output to explicitly signal task completion. We refer to our backbone architecture as MT-ACT+. To ensure peak performance under all experimental conditions, we independently optimise the number of network parameters for each method that uses a BC policy and for each data regime.

Loss Function: Just like the network architecture, the loss function used to train all BC policies was kept consistent. During training, all policies maximize the log-likelihood of demonstration action chunks, i.e.

$$\min_{\theta} \Sigma_{o_i, a_i, l \sim D} \, \pi_\theta \left(a_{i:i+k} | o_i, l \right),$$

with the standard VAE objective which has a reconstruction loss and a term that regularizes the encoder to a Gaussian prior. Here, o_i and $a_{i:i+k}$ are a sampled target object point cloud and an action chunk (see below) and l is the corresponding task description. We further augment this loss by using learned weighting with homoscedastic uncertainty (Kendall & Cipolla, 2017) to automatically learn the weighting between different components of the reconstruction loss.

Additional Demonstration Processing: We further process all demonstrations to be suitable to train policies to interact with objects (used by Ret-BC and BC-BC), and to be suitable for training the MT-ACT+ baseline (see the "Combining Simulated Alignment Trajectories and Demonstrations" subsection below). First, we encode the task descriptions l using CLIP (Radford et al., 2021). Next, to ensure a uniform spatial resolution across demonstrations, we subsample demonstrated trajectories to maintain consistent 1cm distances between consecutive waypoints while preserving important events like gripper state changes. We then compute actions $a_{i:i+k}$ as relative poses between the current end-effector pose and future poses within the prediction horizon k, using angle-axis representation for orientations. Similarly, we compute history action labels as relative poses between the current pose and past poses within the history horizon.

Data Augmentation: Regardless of whether we would like a policy to learn to align, interact or to handle both phases, we apply common augmentation steps whenever an observation-action tuple is sampled during training. To improve robustness to partial occlusions and varied object poses encountered during deployment, we mask out random portions of the target object point cloud. To this end, we perform furthest point sampling followed by nearest neighbour clustering to create 10 clusters, of which we randomly mask 4. We have found this to help during preliminary experiments. And to improve robustness to noise in point clouds and action history labels, we add Gaussian noise to both. Next, we explain how we further process demonstrations to be able to train the BC

architecture to interact with objects, how we simulate end-effector trajectories to train the BC policy to align with objects, and finally, how we combine the processed demonstrations and simulated trajectories to obtain training data for our monolithic baseline MT-ACT+.

To achieve better robustness to covariate shift when learning from limited data to interact with objects, whenever an observation-action tuple is sampled during training, we perturb the end-effector pose within 0.9 cm of its original position and 5 degrees of its original orientation. We then update the corresponding state, first action label, and history label to reflect this perturbation. This augmentation helps the policy become robust to small deviations from the demonstrated trajectories that may occur during deployment and is only feasible because the policy takes as input the target object point cloud expressed in the end-effector frame.

Simulating Alignment Trajectories for Behaviour Cloning: To be able to learn to align the end-effector with target objects, both the alignment BC policy used by BC-Ret and BC-BC, and the MT-ACT+ baseline need trajectories for training that move the end-effector to the first pose of each demonstration. To this end, we simulate 1000 alignment trajectories per demonstration, by sampling starting poses within a 30x80x80 cm cuboid above the robot's taskspace and generating linear trajectories to the end-effector poses at the beginning of demonstrations. We achieve this by simply moving the target object point cloud captured during the first frame of the demonstration in the end-effector frame, while maintaining a fixed distance of 1cm between waypoints. This is only feasible because we use the target object point cloud expressed in the end-effector frame as the input to the policy.

Furthermore, to help the alignment policies used by BC-Ret and BC-BC learn to accurately align with objects, we supplement their training data with additional observation-action pairs near each final alignment pose. For each waypoint in a simulated alignment trajectory, we generate an additional observation-action pair by randomly perturbing the end-effector pose within 1mm-1cm and 0.5-5 degrees of the final alignment pose.

Combining Simulated Alignment Trajectories and Demonstrations: Our monolithic baseline, MT-ACT+, requires training data for both the alignment and interaction phases of demonstrated tasks. As such, we combine the simulated alignment trajectories with demonstrated trajectories to create a dataset of entire manipulation trajectories, adjusting the history and action labels at the boundary between alignment and interaction phases.

A.5 Controlled Experiment Evaluation Procedure

For both experiments (scaling demonstrations per task and scaling number of tasks), we conduct three evaluations per task and average results across all micro skills. For each evaluation, we randomize the object's position within the 80 x 45 cm taskspace, and orientation by ±180 degrees around the vertical axis from the demonstration pose. Success rates are determined through manual evaluation, where an expert observer monitors each rollout and classifies it as successful only if the robot completes the manipulation task.

A.6 Scaling to a Thousand Tasks

Experimental Design: We deliberately designed the experiment to stress-test MT3's capabilities across multiple dimensions:

- Manipulation Complexity: Tasks ranged from simple grasping and placing motions, to tasks requiring precise insertion or complex non-linear trajectories.
- Object Variety: We included particularly challenging items for depth sensors, including semi-transparent and transparent objects like plastic containers and glass cups, highly deformable objects like clothing, reflective metallic objects like toasters, and articulated objects such as cabinets and boxes.
- Environmental Variation: During evaluation, each task execution faced substantial real-world complexity. For testing, we placed 5-20 distractor objects in the scene alongside the target object. Between different evaluation rollouts, we actively varied the lighting conditions by changing both the color and intensity of an LED light source. To further challenge robustness, we randomised object placement anywhere within the workspace with up to

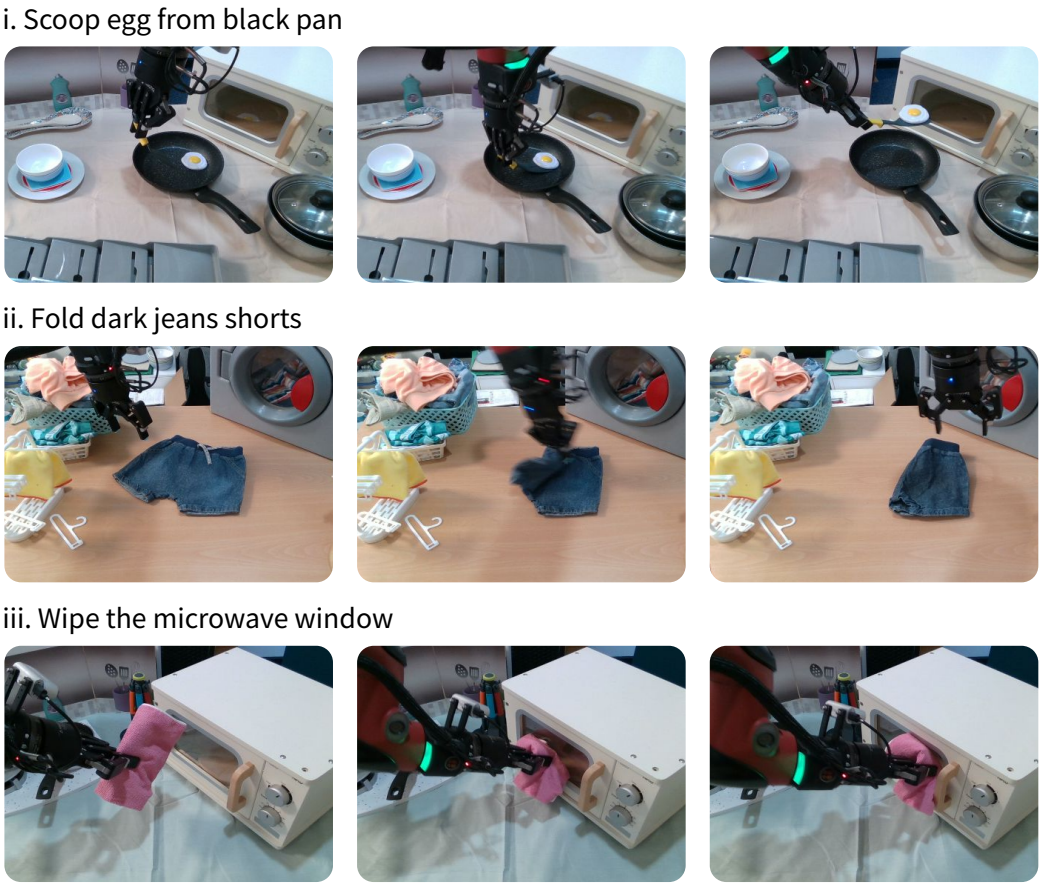

Figure 12: Examples of recorded rollouts from the 1000 tasks experiment.

45-degree rotational variation from demonstration poses. Furthermore, we deliberately changed the colour of the surface on which objects were placed between demonstration and testing phases. Figure 8 illustrates these diverse test environments, showcasing the significant variations in lighting, surface colour, and scene composition that our system was challenged with.

This experimental design represents one of the most comprehensive evaluations of robot manipulation learning to date, combining unprecedented task diversity with challenging real-world conditions - all while maintaining the constraint of single-demonstration learning. Figure 12 showcases example rollouts of from this experiment.

Failure Mode Analysis: To better understand MT3's limitations and identify paths for improvement, we conducted a detailed analysis of failure cases on seen tasks. An expert evaluator assessed each rollout across four key aspects: correct segmentation, exact retrieval, pose estimation success, and motion execution. Figure 13 presents this systematic breakdown of failure modes.

The objective of retrieval is to identify the most suitable demonstration for the test scenario by finding one which performs the same micro skill on a similar object both in appearance and pose. This process however has emerged as the primary challenge, accounting for 22.3% of failures. These occurred most frequently with partially occluded objects or in cases where the relevant variations in object geometry regarded smaller object parts that are therefore harder to discern. While multiple cameras would provide more complete object observations for retrieval, isolating the relevant object part remains challenging.

Segmentation and pose estimation problems each contributed significantly to system failures, at 19.5% and 23.9% respectively. Segmentation challenges arose predominantly with transparent objects and in cluttered scenes with similar-looking items. However, as segmentation models con-

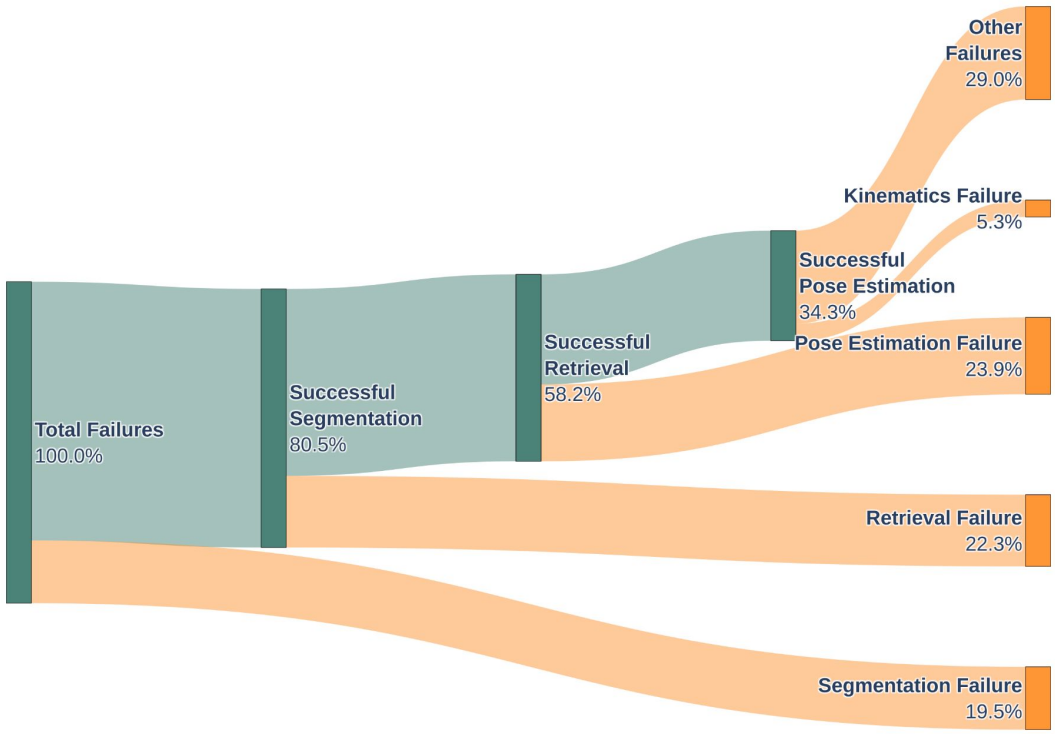

Figure 13: A sankey diagram illustrating MT3's failure modes when evaluated across 1000 seen tasks.

tinue to advance, these issues are expected to diminish over time. Pose estimation proved particularly challenging with drastic changes in pose with respect to the demonstration, as these also result in substantially different partial point clouds due to asymmetric geometries and perspective changes. A multi-camera setup would provide more complete geometric information and reduce these perspective-related challenges.

Notably, issues due to pure motion planning and kinematic did occasionally occur, but they were rare, accounting for only 5.3% of failures. The remaining 29% of failures were mainly from tasks with grasped objects (20.2%), such as insertions or scooping, where the grasped object could have been placed inconsistently between demonstration and deployment. Such failures can be mitigated using the method proposed by Papagiannis et al. (2024), which enables learned skills to generalize across different grasps without additional training. The remainder of failures were difficult to pinpoint, but could include calibration drift and fine-grained misalignment errors.