

# RoboCade: Gamifying Robot Data Collection

Suvir Mirchandani<sup>\*1</sup>, Mia Tang<sup>\*1</sup>, Jiafei Duan<sup>2</sup>, Jubayer Ibn Hamid<sup>1</sup>, Michael Cho<sup>3</sup>, Dorsa Sadigh<sup>1</sup>

**Abstract**— Imitation learning from human demonstrations has become a dominant approach for training autonomous robot policies. However, collecting demonstration datasets is costly: it often requires access to robots and needs sustained effort in a tedious, long process. These factors limit the scale of data available for training policies. We aim to address this scalability challenge by involving a broader audience in a gamified data collection experience that is both accessible and motivating. Specifically, we develop a gamified remote teleoperation platform, RoboCade, to engage general users in collecting data that is beneficial for downstream policy training. To do this, we embed gamification strategies into the design of the system interface and data collection tasks. In the system interface, we include components such as visual feedback, sound effects, goal visualizations, progress bars, leaderboards, and badges. We additionally propose principles for constructing gamified tasks that have overlapping structure with useful downstream target tasks. We instantiate RoboCade on three manipulation tasks—including spatial arrangement, scanning, and insertion. To illustrate the viability of gamified robot data collection, we collect a demonstration dataset through our platform, and show that co-training robot policies with this data can improve success rate on non-gamified target tasks (+16–56%). Further, we conduct a user study to validate that novice users find the gamified platform significantly more enjoyable than a standard non-gamified platform (+24%). These results highlight the promise of gamified data collection as a scalable, accessible, and engaging method for collecting demonstration data. Videos are available at [robocade.github.io](https://robocade.github.io).

## I. INTRODUCTION

In recent years, imitation learning from human demonstrations has become an increasingly popular approach for training robot policies to perform versatile skills [1–4]. A key factor powering these approaches is investment in real-world robot data collection. However, as modern imitation learning methods scale up, so do the requirements of data collection—often on the order of hundreds to thousands of demonstrations for individual tasks [3, 4]. Thus, scaling real-world data collection remains a crucial bottleneck.

Two key challenges underlie this bottleneck: limited access to robots, and limited incentive to engage in data collection. First, robot data collection typically requires physical access to costly robot hardware, restricting participation to a small pool of trained operators and researchers. Second, while remote teleoperation platforms [5, 6] and robot-free platforms [7, 8] have emerged as a promising way to broaden access, they are typically designed with functionality as the primary concern. These systems often do not actively prioritize user engagement in their design. Thus, even when access is no longer a limiting factor, there remains little incentive for



Fig. 1: We develop **RoboCade**, a platform that gamifies the collection of robot demonstration datasets. In standard data collection approaches (*top*), hired operators or researchers collect demonstrations using in-person teleoperation, which requires access to a robot and can be tedious and time-consuming. RoboCade (*bottom*) is a remote data collection platform that integrates gamification into both system and task design, making robot data collection more engaging and accessible to a broader set of users.

broader participation: the data collection process remains tedious and unengaging for most users. As a result, most datasets in academia and industry are collected in-house through dedicated data collection efforts by researchers or hired operators [4, 9–13].

To address the challenges of access and incentive, we propose examining robot data collection through the lens of *gamification*—the integration of game-like elements into non-gaming systems [14]—specifically in the context of remote teleoperation. Gamification has been successfully applied to diverse domains from education to commerce to increase user participation and engagement [15]. In this vein, we propose a framework for gamifying robot data collection by recasting remote teleoperation—typically via a purely utilitarian interface—into an engaging interactive experience. By embedding gamification principles into both system interface and task design, we aim to enable experiences that are not only engaging but also yield useful robot demonstrations.

We instantiate this framework in a web-based platform, RoboCade. Built on top of a 3D-printable GELLO controller [16], which enables intuitive teleoperation at a low-cost, RoboCade allows users to teleoperate real robots remotely, with game mechanics embedded throughout the

<sup>\*</sup>Equal contribution. {suvir,miatang}@cs.stanford.edu

<sup>1</sup>Stanford University; <sup>2</sup>University of Washington; <sup>3</sup>FrodoBots

data collection process. We instantiate this system on 3 manipulation tasks—including spatial rearrangement, insertion, and scanning tasks—that are recast as interactive game-like experiences.

Our contributions are as follows. First, we develop a gamified remote teleoperation platform based on a web-based interface with a GELLO controller, featuring a variety of gamified components. Second, we develop a set of principles for designing gamified robot manipulation tasks that integrate game-like mechanics but retain relevance to downstream target tasks. Third, we co-train policies with data collected by our gamified interface, illustrating the viability of gamification in robot data collection. Our simple co-training recipe combines demonstrations of gamified tasks collected using RoboCade and non-gamified target tasks collected using standard teleoperation. We find that co-training policies with data collected through our gamified platform improves in-distribution success on 3 non-gamified target tasks (+16–56%) with additional boosts in out-of-distribution performance up to 20%. Likewise, we observe benefits to including data from gamified tasks in the context of fine-tuning pre-trained vision-language-action models on target tasks. Finally, we demonstrate through a human subjects study ( $N = 18$ ) that users find our gamified interface more intuitive (+27%), enjoyable (+24%), and motivating (+24%) compared to a non-gamified platform.

## II. RELATED WORK

**Gamification of Data Collection.** Outside of robotics, gamification [14] is a widely studied technique to improve user engagement across domains ranging from education to commerce [15, 17]. Early work on Games-with-a-Purpose (GWAPs) [18] proposed the idea of channeling user interactions in game-like experiences towards collecting data for machine learning tasks such as image captioning [19], object recognition [20], and commonsense reasoning [21]. However, the connection between gamification and robot data collection remains underexplored [22]. Our work aims to bridge the gap, scaling robot data collection through gamifying the experience to involve general users.

**Robot Data Collection Platforms.** A variety of interfaces exist for collecting robot demonstrations. These span four families, each with trade-offs in terms of control fidelity, safety, operator availability, and overall scalability. Kinesesthetic teaching [23] is precise and safe, but is labor-intensive and requires collocation with a robot. Common teleoperation input devices (SpaceMouse [24], gamepads [25], commercial VR controllers [26], AR [7, 27], smartphones [5]) boost throughput and allow remote access, but can be cognitively demanding and latency-sensitive. Wearable end-effector systems (DexCap [28], UMI [8], DexUMI [29]) which map human hand poses to the end-effector can allow for higher dexterity and demonstration rates, but can require custom hardware, calibration, and careful retargeting. Recently, purpose-built puppeteering platforms such as ALOHA [1] and GELLO [16] are designed to streamline data collection while preserving real-robot fidelity. Rather than introducing

a new teleoperation device, we leverage GELLO [16], which enables intuitive and high-fidelity teleoperation of real robot arms at a low-cost. Prior works typically treat the input controller for data collection in a purely utilitarian manner. In this work, we build a remote gamified data collection system using GELLO, enabling us to study user engagement in robot data collection, while maintaining real-robot fidelity and removing the requirement for proximity to a real robot.

**Crowdsourcing Robot Data.** Crowdsourcing has emerged as a scalable avenue for robot data collection. Early systems such as RoboTurk [5] crowdsource demonstrations via a smartphone teleoperation interface, enabling remote workers to control robots. More recent efforts extend this paradigm to in-the-wild settings—for example, remotely teleoperated cars to harvest navigation trajectories [30] and low-cost hardware that brings data collection into homes and labs (e.g., SO-100/101 robots via the LeRobot platform [31]). Recent work [32] studies crowdsourcing of in-person robot demonstrations in public spaces, but requires deployment of physical robots, limiting scalability. Other lines of work use crowds to guide exploration in real-world RL [33] or to collect interaction data through higher-level abstractions [34], trading off low-level control for accessibility. A common limitation is reliance on extrinsic incentives, such as monetary or token-based payments, which can cap long-term scale and sustainability. In contrast, RoboCade explores gamification of remote data collection, harnessing aspects such as intrinsic motivation and competition, with the potential to enable scalable crowdsourcing without direct monetary incentives.

## III. ROBOCADE

We develop RoboCade: a platform that gamifies the collection of robot demonstration datasets. In [Section III-A](#), we motivate the need for an accessible and engaging data collection platform in the context of imitation learning. In [Section III-B](#), we reframe robot data collection as a gamified experience and establish desiderata for a platform that enables gamified data collection. Next, in [Section III-C](#) and [Section III-D](#), we describe how we achieve these goals through a combination of system design and task design. We instantiate this platform on three tasks ([Section III-E](#)). In [Section IV](#), we validate RoboCade as a data collection platform via policy learning experiments and a human subjects study.

### A. Problem Setting: Data Collection for Imitation Learning

Imitation learning (IL) aims to learn the behavior of an expert policy  $\pi_E$  performing a task  $\tau$ . IL is often formulated as behavior cloning, where a policy  $\pi_\theta : \mathcal{O} \rightarrow \mathcal{A}$  parameterized by  $\theta$  is trained using supervised learning from a dataset  $\mathcal{D}_\tau = \{\xi_1, \dots, \xi_N\}$  composed of  $N$  expert demonstrations  $\xi_i$ . Each expert demonstration  $\xi_i$  is a sequence of observation-action transitions  $\{(o_0, a_0), \dots, (o_{T_i}, a_{T_i})\}$  with  $o_i \in \mathcal{O}$ ,  $a_i \in \mathcal{A}$ .

In practice, the costs of collecting  $\mathcal{D}_\tau$  can be substantial due to the human effort involved. These datasets are often collected through long and tedious teleoperation sessions in which operators repeatedly collect demonstrations for specific tasks. As a result, the size and composition of



Fig. 2: **System and Task Design Overview.** RoboCode uses a GELLO controller [16] with a web-based interface to enable remote teleoperation of real robots. We embed gamification principles into the design of the system and tasks. (*Top*) Into the system, we incorporate visual and auditory feedback and rewards, challenge and goal visual cues, identity and progression, and social engagement. (*Bottom*) Into the design of tasks, we incorporate narrative, goal diversity, an appropriate level of challenge, and overlapping skills with relevant downstream tasks.

$\mathcal{D}_\tau$  is closely tied to access to operators, availability of robot hardware, and any external incentive (e.g., monetary payment) for collecting data. In this work, we develop a platform that gamifies robot data collection, wherein a broader population can be engaged in data collection through a combination of a gamified remote teleoperation interface and design of engaging tasks.

### B. Gamifying Robot Data Collection

While prior work has demonstrated the effectiveness of gamification across domains [15], applying it to robot data collection introduces unique challenges. First, the system must foster intrinsic motivation through engaging gameplay, without relying on users to care about the data itself, broadening the pool of potential contributors. Second, it must support smooth remote teleoperation while concurrently streaming gamified interface elements with minimal latency. Finally, the system should leverage low-cost teleoperation controllers to enable deployment beyond research labs and increase accessibility for a broader audience. From these considerations, we derive three goals for our system:

- *Engaging.* The system should provide a gamified experience that is intuitive, engaging, and motivating for users.
- *Remote-friendly.* The system should support low-latency, smooth teleoperation for users across diverse locations.
- *Low-cost.* The system should be inexpensive to build and maintain, enabling users to access with minimal resources.

**Operators as Players.** We reconceptualize the role of robot operators by framing them as *players*. Rather than treating teleoperation as a purely functional means of issuing robot commands to collect demonstration data, our system invites users to engage as players in a gamified experience—participating for gameplay rather than solely for data collection. This reframing translates manipulation tasks into interactive challenges. This lowers barriers for broader

participation by enabling playful goal-driven interaction, reducing the need for external incentives.

**Episodes as Attempts.** We frame each teleoperation episode as an *attempt* within a game-like experience. This framing situates demonstrations in a familiar cycle of trial and retry, where users are encouraged to make progress across multiple attempts rather than viewing a single episode as final.

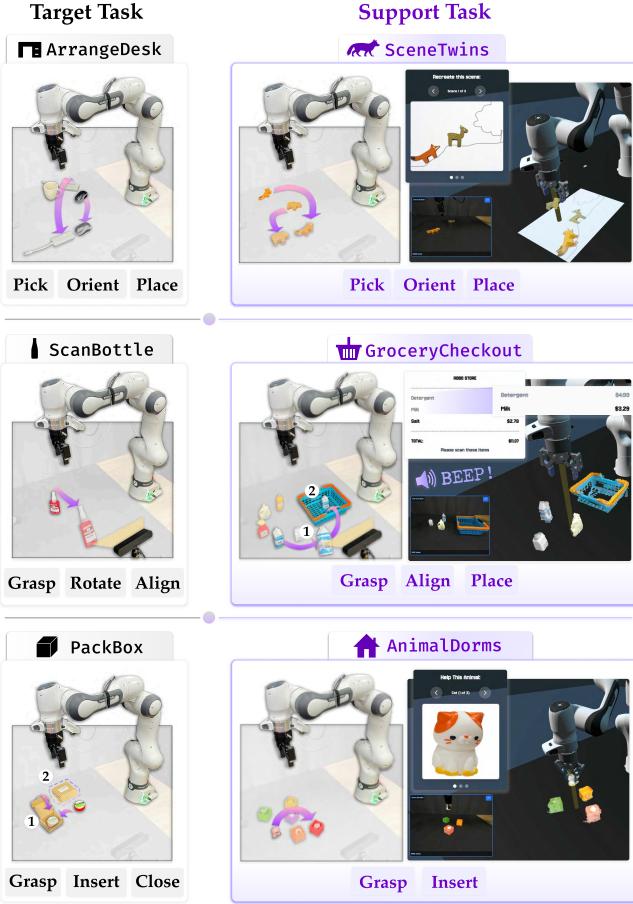
**Teleoperation as Gameplay.** We frame robot teleoperation as a gameplay experience by incorporating established game design elements and curating tasks with gamified structures. The following subsections detail the specific components through which this transformation is realized.

### C. System Design: Gamified Remote Teleoperation

We integrate a specific set of game design elements guided by four core design principles, aimed at fostering engagement from a general audience and, in turn, enabling more scalable robot data collection.

**SD1. Feedback & Reward.** Timely feedback on user actions is crucial to engagement [35]. We incorporate several forms of feedback to reinforce progress—for example, visually highlighting the robot gripper in the interface as a confirmation of a grasp, and rewarding task completion with celebratory visual and auditory effects (Fig. 2).

**SD2. Challenge & Goal Cues.** The system introduces visual cues for challenges and goals to sustain engagement. Each task is bounded by a predefined time limit that is appropriately calibrated yet still presents a challenge for non-expert users. This time limit is visualized with a progress bar that fills as time elapses (Fig. 2). The goals for each task are visualized in ways tailored to the task itself. For instance, in our scanning task, the object list which the user must complete is presented as a grocery store receipt (Fig. 2). In our rearrangement task, a scene that the user must recreate with objects is virtually projected onto the table surface.



**Fig. 3: Task Design.** We show 3 pairs of support tasks (*right*) and target tasks (*left*). Rearrangement (*top*): Both tasks involve moving 2 objects; ArrangeDesk focuses on organizing electronics, while SceneTwins uses animal blocks with diverse layouts and a virtual goal overlay. Scanning (*middle*): both tasks involve grasping and aligning objects for scanning; ScanBottle requires more precision due to the object geometry and randomized orientations, while GroceryCheckout adds object variety, a grocery checkout narrative, sound effects, and a basket placement step. Insertion (*bottom*): both tasks require packing, but PackBox adds complexity by closing a box lid while AnimalDorms adds narrative and object variety.

**SD3. Identity & Progression.** Establishing a sense of personal identity within digital systems is a key driver of sustained engagement and participation [36]. In our platform, users create a persistent profile with a username and avatar that is visible across sessions and displayed on every page, giving them a distinct RoboCade identity. Game participation contributes to a point system that unlocks badges as users engage in more episodes and games, providing a lightweight progression structure that can reinforce longer-term involvement.

**SD4. Social Engagement.** Social engagement, e.g., through competition among participants, is an important aspect of engagement in games [37]. We incorporate a public leaderboard which ranks users by cumulative game points. In addition, each game session generates a unique shareable URL, enabling spectators to observe sessions in real-time.

#### D. Task Design: Support Tasks for Co-Training

In this section, we propose principles for designing tasks for gamified remote data collection. Driving these principles

are two main considerations: tasks should be engaging but nevertheless produce useful data for non-gamified tasks. To this end, we approach task design as a transformation from a downstream *target task*  $\tau$  into a gamified *support task*  $\tau'$ . In Section III-E, we provide 3 instantiations of support tasks.

**TD1. Narrative.** Robot data collection tasks are typically specified through functional language descriptions—e.g., “put object A at location B” [12]. To increase immersion, we embed the same action sequence within a narrative or theme [38], supported via textual and visual prompts. For example, inserting toy objects into a box can be reframed as “helping a character find its home” (Fig. 2), inviting users to engage with the task as part of a story.

**TD2. Goal Diversity.** Repeating the same objective across trials can quickly lead to disengagement. We introduce variations in task goals while preserving underlying manipulation skills, transforming repetitive actions into distinct challenges. For instance, grasping skills can be practiced with different objects, or rearrangement skills can feature different spatial goals (Fig. 2), adding novelty to each episode.

**TD3. Challenge.** Tasks risk being either too trivial to sustain engagement or too difficult for novices to attempt. Guided by the concept of flow [39], we calibrate the difficulty of support tasks to be non-trivial yet attainable through practice.

**TD4. Overlapping Skills.** In a co-training paradigm, numerous factors can affect transfer from support tasks to downstream tasks [40–42]. When designing support tasks, we incorporate at least one shared manipulation skill between target and support tasks: for example, grasping objects of similar geometries or inserting items into a small box (Fig. 2); the specific low-level motion can differ between target and support tasks, but the high-level motion is similar.

#### E. Task Details

We now apply these principles to the design of 3 task classes, shown in Fig. 3, and instantiate them in RoboCade. Our task design is guided by 3 target tasks (ArrangeDesk, ScanBottle, PackBox), resulting in 3 support tasks (SceneTwins, GroceryCheckout, AnimalDorms). In Section IV-A, we demonstrate that co-training on support tasks can be beneficial for learned policies on target tasks.

**Rearrangement (Fig. 3; top).** The ArrangeDesk target task requires sequentially picking up two pieces of electronics—a USB adapter and a computer mouse, placed in an arbitrary configuration on one side of a desk, alongside a mug—and arranging them in a straight line on the other side of the desk. This task requires precision in grasping (the adapter lies flat on the table at about 1 cm in height, and the mouse has a curved surface) as well as placement to arrange the items in a line. As a support task, we design the SceneTwins task: users rearrange two animal blocks to recreate a given scene (TD1). The goal locations are more diverse (TD2) than in ArrangeDesk and are presented to the user with a scene sketch virtually overlaid on the table (SD2). We vary the specific scene between episodes, and matching the scene exactly with the overlay introduces challenge (TD3). At a high-

level, this task requires similar skills of grasping, orienting, and arranging objects as in the ArrangeDesk task (TD4).

**Scanning (Fig. 3; middle).** The ScanBottle target task involves picking up a bottle of threadlocker solution and orienting the barcode on the bottle in front of a scanning camera. We design a support task GroceryCheckout, which is a multi-step task framed as checking out groceries from a receipt within the given time (TD1, SD2). GroceryCheckout requires picking up a toy grocery item, bringing it to the scanner, and placing it in a toy grocery basket. Upon successfully scanning, the interface plays an auditory beep, checks off the item from the receipt, and displays celebratory virtual confetti (SD1). This task entails shared skills from ScanBottle, including grasping and aligning with the scanner (TD4). GroceryCheckout also contains the extra step of placing in the grocery basket. To make the task easier, the motion required is simpler in that it does not require a significant amount of reorientation, and grasping the objects is more straightforward than grasping the threadlocker bottle which has a narrow cap (TD3). For diversity, GroceryCheckout uses a variety of objects compared to ScanBottle which uses just one (TD2). ScanBottle can be regarded as a specialized version of one skill in GroceryCheckout, focusing on a harder object, without the diversity of objects and gamified effects.

**Insertion (Fig. 3; bottom).** PackBox is a multi-step task which requires picking up a roll of electrical tape, inserting it in a cardboard box, and closing the box shut. This task requires precise motion, particularly when closing the lid along its joint. To provide a support task with a more appropriate level of difficulty, we design the AnimalDorms support task, which involves picking up an animal toy and inserting it into a small toy box of a matching color (TD3). This task is framed as “returning an animal to its home” (TD1). This task shares the skills of grasping and insertion with PackBox (TD4). We introduce diversity into this task by (a) varying the animal toy in each episode and (b) randomizing the location of the animal toy and houses across a larger range of initial states than in PackBox (TD2).

#### F. System Architecture and Implementation Details

We implement RoboCade using the Franka FR3 robot arm. For parity with the DROID platform [12], we use two third-person ZED2 cameras and one ZED Mini egocentric camera. As the input device, we use GELLO [16] with joint impedance control. We select GELLO as it is low-cost, buildable by hobbyists, and allows for intuitive joint-space control, naturally obeying kinematic constraints and avoiding self-collisions; additionally, [16] finds that novice users prefer GELLO compared to Cartesian control with a 3D mouse or a VR controller. For safety, we prevent the robot’s joints from exiting workspace bounds to avoid collisions with the table and cameras. We use ZeroMQ [43] for message passing and WebSockets for transmitting data between the backend and frontend. Within the main viewer, a live point cloud of objects in the scene is displayed alongside a virtual rendering of the robot as well as additional visual

effects. The camera of the 3D viewer moves based on the user’s action to provide a sense of depth. We implement the RoboCade frontend using Next.js [44] and Three.js [45] for 3D rendering. Additional implementation details are given in the Appendix.

## IV. EXPERIMENTS

In our experiments, we evaluate RoboCade as a data collection platform in two main aspects. First, in [Section IV-A](#) and [Section IV-B](#), we evaluate whether data collected from gamified support tasks can effectively improve downstream policy learning on non-gamified target tasks through co-training. Second, in [Section IV-C](#), we evaluate how usable and engaging the system is via a human subjects study.

### A. Policy Learning: Co-training with Gamified Data

First, we investigate the usefulness of gamified data in policy learning. We examine how data collected on support tasks can be a viable source of co-training data for target tasks—reducing the reliance on target-specific data.

**Experiment Setup.** For each pair of target and support tasks, we collect a pair of datasets ( $\mathcal{D}_\tau$ ,  $\mathcal{D}_{\tau'}$ ). We collect  $\mathcal{D}_\tau$  through standard in-person teleoperation with GELLO, and  $\mathcal{D}_{\tau'}$  through the RoboCade platform. For ArrangeDesk and ScanBottle,  $\mathcal{D}_\tau$  consists of 80 demonstrations, and for PackBox,  $\mathcal{D}_\tau$  consists of 150 demonstrations. For each support task (SceneTwins, GroceryCheckout, and AnimalDorms),  $\mathcal{D}_{\tau'}$  contains 200 demonstrations. For each task pair, we compare policy performance between training on target data  $\mathcal{D}_\tau$  alone to co-training with gamified support data. Training only on  $\mathcal{D}_\tau$  constitutes the *Target Only* setting. Co-training on  $\mathcal{D}_\tau$  and  $\mathcal{D}_{\tau'}$  constitutes the *Co-train* setting.

**Training Details.** We train policies with Diffusion Policy [46] using third-person and egocentric camera images and end-effector position as the observation, and absolute joints as actions. We follow the U-Net architecture from [46] with action chunks of size 16 and execution horizon of 8. For co-training, we sample batches with a 50%-50% split between data sources. For fair comparison, all policies are trained for the same number of steps (300K). Policies operate at 12Hz.

**Evaluation Protocol.** For each task and training condition, we perform 25 trials. We randomize the initial poses of objects at each trial, and use image overlays to maintain consistency in initial poses when evaluating different policies on the same task to minimize evaluation variance [47]. We record success rate for each task at different stages.

**Results.** [Fig. 4](#) summarizes our results. We find that co-training with gamified data from support tasks improves policy performance across all target tasks, with the magnitude and type of improvement varying by task.

**ArrangeDesk.** This target task has multiple steps: two pick-and-place motions to move the USB adapter and the computer mouse away from the mug and line them up at the opposing side of the table. The relative positions and orientations of the mug, adapter, and mouse are randomized at one side of the table. In the corresponding support task SceneTwins, two animal blocks are randomized in position

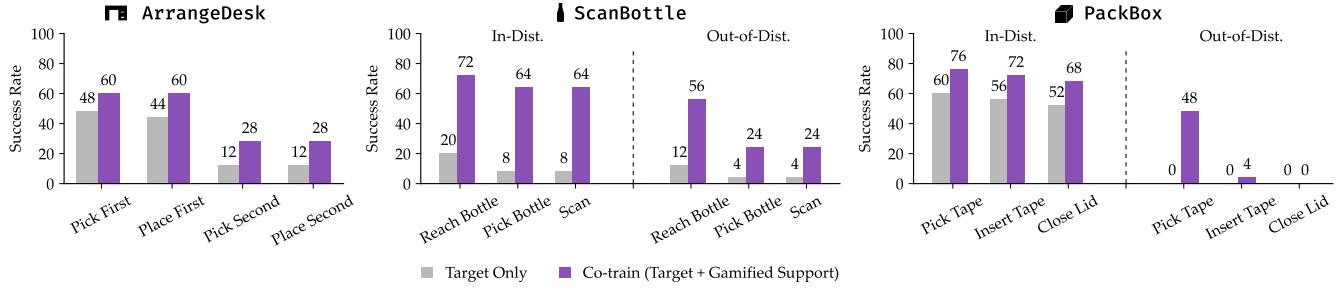


Fig. 4: **Co-training with Gamified Data.** For our 3 target tasks, we compare the performance of Diffusion Policy [46] when trained only on target task data (*Target Only*) versus co-trained with gamified support tasks (*Co-train*), with a fixed training budget. For each task and training condition, we perform 25 trials and report staged success rate (%). Co-training improves success rate on all 3 tasks for in-distribution conditions (*In-Dist.*). For ScanBottle and PackBox, we additionally evaluate on out-of-distribution initial configurations (*Out-of-Dist.*) and find that co-training improves generalization.

and orientation (in the same general region on the table) but their final position varies based on the goal scene. With only 80 demonstrations in  $\mathcal{D}_\tau$ , the target-only policy for ArrangeDesk achieves only 12% overall success, but co-training with SceneTwins data increases the success to 28%. Qualitatively, we observe that successful rollouts are higher quality—i.e., with more aligned grasps and less retrying.

**ScanBottle.** In this task, the threadlocker bottle’s initial position and orientation are randomized within a roughly 1.5-square-foot region. In GroceryCheckout, the items to be scanned are toy grocery items (e.g., a toy carton of milk). GroceryCheckout entails a similar basic motion to ScanBottle (grasping an object and aligning it in front of the scanner camera), but scanning the threadlocker bottle requires more precision to grasp from the top due to its longer and thinner cap. GroceryCheckout involves an extra step of placing the object in a grocery basket. The region of initial object placements in GroceryCheckout partially overlaps with that of ScanBottle, but extends to other areas of the table. Interestingly, we find that co-training with GroceryCheckout substantially improves in-distribution performance (>50%) on ScanBottle, even including regions of the ScanBottle initial distribution that are not covered by GroceryCheckout. We also evaluate ScanBottle in out-of-distribution initial states—regions that are not covered by ScanBottle’s target data—and find that policy performance improves by approximately 20%. We find that co-training with GroceryCheckout helps substantially at the stage of localizing the threadlocker bottle precisely enough to grasp.

**PackBox.** This multi-step task is challenging, requiring first grasping the roll of electrical tape, then inserting it into the box, and lastly closing the lid shut. Thus, we collect a larger set of target data (150 demonstrations). AnimalDorms has shared skills with this task—picking up a small toy and inserting it into a box. AnimalDorms’s animal initial positions are more diverse than that of PackBox. We see that co-training with AnimalDorms provides a 16% improvement on in-distribution performance, particularly at the stage of picking up the tape roll. Qualitatively, we observe that grasps are more precise after co-training (being aligned closer to the center axis of the roll of tape). We additionally perform out-of-distribution evaluations on PackBox, and find that with co-training, the learned policy PackBox is able to generalize

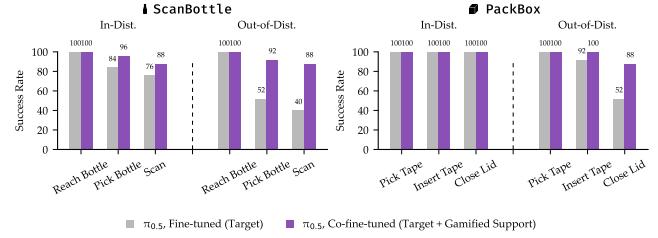


Fig. 5: **VLA Co-fine-tuning with Gamified Data.** We compare the performance of a version of  $\pi_{0.5}$  [48] fine-tuned on the DROID dataset [12] when further fine-tuned on target task data versus co-fine-tuned on both target task and gamified support task data, with a fixed budget of training steps. For each task and training condition, we perform 25 trials. Co-fine-tuning on gamified support task data improves performance in out-of-distribution initial configurations (*Out-of-Dist.*) while matching or exceeding performance in in-distribution conditions (*In-Dist.*).

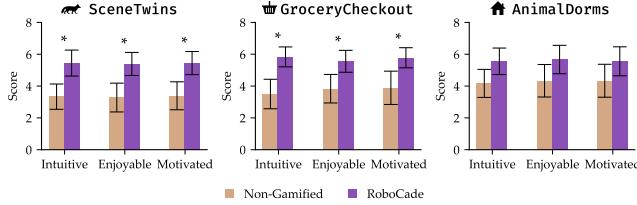
the pick subtask to locations unseen in the PackBox data. Overall, these results validate the idea that gamified support tasks can produce beneficial data for specialized policies of more realistic target tasks.

#### B. Policy Learning: VLA Co-fine-tuning with Gamified Data

Next, we examine whether the utility of gamified data holds in the context of larger, pre-trained vision-language-action (VLA) models.

**Experiment Setup.** We begin with a version of  $\pi_{0.5}$  [48] fine-tuned on the DROID dataset [12] and further fine-tune it on  $\mathcal{D}_\tau$  or co-fine-tune it on  $\mathcal{D}_\tau$  and  $\mathcal{D}_{\tau'}$ . For fair comparison, we perform model selection for each task on the target-only condition to determine the budget of training steps used for both conditions. Third-person and egocentric camera images are provided as the observation. We use action chunks of size 16 and an execution horizon of 10. For evaluation, we use the same initial configurations from Section IV-A for in-distribution and out-of-distribution conditions, with 25 trials per task and condition.

**Results.** Fig. 5 summarizes our results. Relative to the policies from Section IV-A, both the fine-tuned and co-fine-tuned  $\pi_{0.5}$  models achieve substantially higher success rates in both in-distribution and out-of-distribution settings. Both the fine-tuned and co-fine-tuned models can robustly localize and reach the initial object of interaction. Compared to the fine-tuned model, we observe that co-fine-tuning with



**Fig. 6: User Study Subjective Results.** We report user rankings for our RoboCade system compared to a non-gamified remote teleoperation interface. The GELLO controller is used in both conditions. Users tend to find the gamified interface more intuitive, enjoyable, and motivating. Error bars indicate 95% confidence intervals. \* denotes significance at  $p < 0.05$  under a Wilcoxon signed-rank test.

support task data further improves in-distribution performance in ScanBottle (+12%) and maintains performance in PackBox. Co-fine-tuning also yields gains in generalization to out-of-distribution configurations, improving success on ScanBottle (+48%) and PackBox (+36%). Qualitatively, we find that for ScanBottle, the co-fine-tuned policy achieves more stable grasps at the picking stage, and for PackBox, the co-fine-tuned policy is more precise at placing the tape in the middle of the box (rather than at the edge) which is necessary to close the lid. In sum, these results demonstrate that gamified tasks, designed following the principles in Section III-D, can provide useful data for downstream policies of target tasks.

### C. User Study

Next, we evaluate user engagement and the usability of RoboCade through a human subjects study with  $N = 18$  novice users coming from diverse backgrounds, including 7 users located in a different U.S. state than the robot.

**Participants and Procedure.** We conduct a within-subjects study with 18 participants (6 female and 12 male, ages 20–36) performing remote teleoperation. For each participant, the study takes approximately 45 minutes to 1 hour. After providing informed consent, each user participates in a familiarization task involving pick-and-place movements, with RoboCade and a baseline remote teleoperation system using the GELLO. The goal of the familiarization task is to allow the user to understand the mechanics of each interface and how to perform teleoperation with the GELLO. Each user is allowed up to 5 minutes of familiarization time with each system. Next, the user performs 3 tasks (SceneTwins, AnimalDorms, GroceryCheckout). The user performs each task 3 times with both systems, with up to 3 attempts allowed per trial, and a limit of 5 minutes per combination of task and condition. We counterbalance the order of tasks and conditions.

**Robot Control Method.** We compare RoboCade to a vanilla remote teleoperation interface (a multi-camera view without gamified components). To control for teleoperation input device, we use the same GELLO controller for both conditions.

**Dependent Measures.** After each task, users complete a qualitative survey regarding how intuitive they found each condition, their level of enjoyment, and their level of motivation to continue the session, on a Likert scale from 1–7.

**Results.** We report subjective survey results in Fig. 6. We run Wilcoxon signed-rank tests with Holm-Bonferroni correction to compare user ratings for RoboCade and the non-gamified system. Users report that they prefer the gamified RoboCade system as being more intuitive (+27%), enjoyable (+24%), and leading to a higher level of motivation (+24%). This effect is more pronounced for SceneTwins and GroceryCheckout ( $p < 0.05$ , Holm-corrected) than for AnimalDorms, where the effects are marginally significant, which we hypothesize is because AnimalDorms is a shorter and easier task overall. Participants additionally have a higher task completion rate with the gamified RoboCade compared to the baseline system (77.1% versus 69.7%). Additionally, users prefer the gamified system to the baseline in terms of aggregate perceived usability, with an SUS score [49] of  $71.8 \pm 20.4$  compared to  $51.4 \pm 22.4$ .

## V. DISCUSSION

We propose an approach for scaling robot data collection by designing gamified remote teleoperation experiences that engage general users, expanding the pool of operators to include users motivated by interactive gameplay. Specifically, we introduce RoboCade, a platform that applies gamification principles to both system design and task design. Through policy learning experiments, we demonstrate the viability of using gamified task data to support policy learning and potentially offload some of the cost of collecting target-specific data. Through a human subjects study, we validate the system as an engaging data collection platform, significantly enhancing participant enjoyment compared to a non-gamified baseline.

**Limitations and Future Work.** Our current work has limitations that open up several avenues for future work.

**Data Quality Incentivization.** The general population has largely untapped potential for robot data collection. However, quality control can be challenging in the context of novice users. Future work can extend RoboCade to incorporate data quality metrics [50–52]—e.g., through a point system that incentivizes high quality trajectories. Notably, two of our study’s best performing users reported having significant experience playing video games despite a lack of teleoperation experience. This suggests an opportunity to tap into the gaming community for scaling up robot data collection by embedding robot control into longer-form games.

**In-depth Study of Game Elements.** RoboCade adapts a number of gamification components to robot data collection. While the study of the fine-grained effect of individual game components on users’ behavioral and psychological outcomes [15, 53] is outside the scope of this work, our platform can enable more in-depth study of these components in the context of robot data collection. Future work can also study long-term interactions among general users with the platform to understand user retention and novelty effects [15, 17, 54], as well as to scale up data collection.

**Learning from Play.** This work focuses on collecting task-directed interactions and utilizing them for co-training. Another avenue for gamified robot data collection is to

endow users with more creativity and agency in their interactions, and collect task-agnostic play data; such data could be used in learning paradigms beyond co-training (e.g., training goal-conditioned methods or world models) [55, 56]. Overall, the intersection of gamification and robotics presents a rich set of research directions which we hope RoboCade can help enable.

#### ACKNOWLEDGEMENTS

We acknowledge funding support from Toyota Research Institute, NSF Award 1941722 and 2006388, ONR YIP and Grant W911NF2210214. We are grateful for useful discussions and feedback from Aaron Tung, Jenn Grannen, Jensen Gao, Shuang Li, Aykut Onol, and Mark Zolotas.

#### REFERENCES

- [1] T. Zhao, V. Kumar, S. Levine, and C. Finn, “Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware,” in *Robotics: Science and Systems (RSS)*, 2023.
- [2] M. Kim et al., “OpenVLA: An Open-Source Vision-Language-Action Model,” in *Conference on Robot Learning (CoRL)*, 2024.
- [3] T. Z. Zhao et al., “ALOHA Unleashed: A Simple Recipe for Robot Dexterity,” in *Conference on Robot Learning (CoRL)*, 2025.
- [4] K. Black et al., “ $\pi_0$ : A Vision-Language-Action Flow Model for General Robot Control,” *arXiv*, 2024.
- [5] A. Mandlekar et al., “RoboTurk: A Crowdsourcing Platform for Robotic Skill Learning through Imitation,” in *Conference on Robot Learning (CoRL)*, 2018.
- [6] S. Dass et al., “TeleMoMa: A Modular and Versatile Teleoperation System for Mobile Manipulation,” *arXiv*, 2024.
- [7] J. Duan, Y. R. Wang, M. Shridhar, D. Fox, and R. Krishna, “AR2-D2: Training a Robot Without a Robot,” *Conference on Robot Learning (CoRL)*, 2023.
- [8] C. Chi et al., “Universal Manipulation Interface: In-The-Wild Robot Teaching Without In-The-Wild Robots,” in *Robotics: Science and Systems (RSS)*, 2024.
- [9] S. Dasari et al., “RoboNet: Large-Scale Multi-Robot Learning,” in *Conference on Robot Learning (CoRL)*, 2019.
- [10] A. Brohan et al., “RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control,” *arXiv*, 2023.
- [11] H. Walke et al., “BridgeData V2: A Dataset for Robot Learning at Scale,” in *Conference on Robot Learning (CoRL)*, 2023.
- [12] A. Khazatsky et al., “DROID: A Large-Scale In-The-Wild Robot Manipulation Dataset,” in *Robotics: Science and Systems (RSS)*, 2024.
- [13] J. Lee et al., “MolmoAct: An Action Reasoning Model that Reasons in 3D space,” *arXiv*, 2025.
- [14] S. Deterding, D. Dixon, R. Khaled, and L. Nacke, “From Game Design Elements to Gamefulness: Defining “Gamification”,” in *International Academic MindTrek Conference: Envisioning Future Media Environments*, 2011.
- [15] J. Koivisto and J. Hamari, “The Rise of Motivational Information Systems: A Review of Gamification Research,” *International Journal of Information Management*, 2019.
- [16] P. Wu, Y. Shentu, Z. Yi, X. Lin, and P. Abbeel, “GELLO: A General, Low-cost, and Intuitive Teleoperation Framework for Robot Manipulators,” in *International Conference on Intelligent Robots and Systems (IROS)*, 2024.
- [17] J. Hamari, J. Koivisto, and H. Sarsa, “Does Gamification Work? – A Literature Review of Empirical Studies on Gamification,” in *Hawaii International Conference on System Sciences*, 2014.
- [18] L. Von Ahn and L. Dabbish, “Designing Games with a Purpose,” *Communications of the ACM*, 2008.
- [19] L. von Ahn and L. Dabbish, “Labeling Images with a Computer Game,” in *Conference on Human Factors in Computing Systems (CHI)*, 2004.
- [20] L. von Ahn, R. Liu, and M. Blum, “Peekaboom: A Game for Locating Objects in Images,” in *Conference on Human Factors in Computing Systems (CHI)*, 2006.
- [21] L. von Ahn, M. Kedia, and M. Blum, “Verbosity: A Game for Collecting Common-Sense Facts,” in *Conference on Human Factors in Computing Systems (CHI)*, 2006.
- [22] I. Leite, W. Ahlberg, A. Pereira, A. Sestini, L. Gisslén, and K. Tollmar, “A Call for Deeper Collaboration Between Robotics and Game Development,” in *IEEE Conference on Games (CoG)*, 2025.
- [23] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, “A Survey of Robot Learning from Demonstration,” *Robotics and Autonomous Systems*, 2009.
- [24] Y. Zhu, A. Joshi, P. Stone, and Y. Zhu, “VIOLA: Imitation Learning for Vision-Based Manipulation with Object Proposal Priors,” in *Conference on Robot Learning (CoRL)*, 2023.
- [25] Y. R. Wang et al., “RoboEval: Where Robotic Manipulation Meets Structured and Scalable Evaluation,” *arXiv*, 2025.
- [26] A. Brohan et al., “RT-1: Robotics Transformer for Real-World Control at Scale,” in *arXiv*, 2022.
- [27] J. Wang, C.-C. Chang, J. Duan, D. Fox, and R. Krishna, “EVE: Enabling Anyone to Train Robot using Augmented Reality,” *ACM Symposium on User Interface Software and Technology (UIST)*, 2024.
- [28] C. Wang, H. Shi, W. Wang, R. Zhang, L. Fei-Fei, and C. K. Liu, “DexCap: Scalable and Portable Mocap Data Collection System for Dexterous Manipulation,” in *Robotics: Science and Systems (RSS)*, 2024.
- [29] M. Xu et al., “DexUMI: Using Human Hand as the Universal Manipulation Interface for Dexterous Manipulation,” *arXiv*, 2025.

- [30] N. Hirose, L. Ignatova, K. Stachowicz, C. Glossop, S. Levine, and D. Shah, “Learning to Drive Anywhere with Model-Based Reannotation,” *arXiv*, 2025.
- [31] M. Shukor et al., “SmolVLA: A Vision-Language-Action Model for Affordable and Efficient Robotics,” *arXiv*, 2025.
- [32] S. Mirchandani et al., “RoboCrowd: Scaling Robot Data Collection through Crowdsourcing,” in *International Conference on Robotics and Automation (ICRA)*, 2025.
- [33] M. Balsells, M. M. L. Torné, Z. Wang, S. Desai, P. Agrawal, and A. Gupta, “Autonomous Robotic Reinforcement Learning with Asynchronous Human Feedback,” in *Conference on Robot Learning (CoRL)*, 2023.
- [34] M. J.-Y. Chung, M. Forbes, M. Cakmak, and R. P. Rao, “Accelerating Imitation Learning through Crowdsourcing,” in *International Conference on Robotics and Automation (ICRA)*, 2014.
- [35] P. Sweetser and P. Wyeth, “GameFlow: A Model for Evaluating Player Enjoyment in Games,” *Computers in Entertainment (CIE)*, 2005.
- [36] Y. Yang, J. Zhang, and T. Gao, “How do Username and Avatar Affect People’s Engagement with Native Advertising on Social Media: From the Self-Disclosure Perspective,” *Psychology & Marketing*, 2024.
- [37] B. Medler and B. Magerko, “Analytics of Play: Using Information Visualization and Gameplay Practices for Visualizing Video Game Data,” *Parsons Journal for Information Mapping*, 2011.
- [38] D. R. Flatla, C. Gutwin, L. E. Nacke, S. Bateman, and R. L. Mandryk, “Calibration Games: Making Calibration Tasks Enjoyable by Adding Motivating Game Elements,” in *ACM Symposium on User Interface Software and Technology (UIST)*, 2011.
- [39] M. Csikszentmihalyi, “The Flow Experience and its Significance for Human Psychology,” in *Optimal Experience: Psychological Studies of Flow in Consciousness*. Cambridge University Press, 1988, pp. 15–35.
- [40] J. Gao, A. Xie, T. Xiao, C. Finn, and D. Sadigh, “Efficient Data Collection for Robotic Manipulation via Compositional Generalization,” in *Robotics: Science and Systems (RSS)*, 2024.
- [41] A. Maddukuri et al., “Sim-and-Real Co-Training: A Simple Recipe for Vision-Based Robotic Manipulation,” in *Robotics: Science and Systems (RSS)*, 2025.
- [42] V. Saxena et al., “What Matters in Learning from Large-Scale Datasets for Robot Manipulation,” *International Conference on Learning Representations (ICLR)*, 2025.
- [43] *ZeroMQ: An Open-Source Universal Messaging Library*, [zeromq.org](https://zeromq.org).
- [44] *Next.js: The React Framework*, [nextjs.org](https://nextjs.org).
- [45] *Three.js – JavaScript 3D Library*, [threejs.org](https://threejs.org).
- [46] C. Chi et al., “Diffusion Policy: Visuomotor Policy Learning via Action Diffusion,” in *Robotics: Science and Systems (RSS)*, 2023.
- [47] TRI LBM Team, “A Careful Examination of Large Behavior Models for Multitask Dexterous Manipulation,” *arXiv*, 2025.
- [48] K. Black et al., “ $\pi_{0.5}$ : a Vision-Language-Action Model with Open-World Generalization,” in *Conference on Robot Learning (CoRL)*, 2025.
- [49] J. Brooke et al., “SUS - A Quick and Dirty Usability Scale,” *Usability Evaluation in Industry*, 1996.
- [50] Y. Zhang et al., “SCIZOR: A Self-Supervised Approach to Data Curation for Large-Scale Imitation Learning,” *arXiv*, 2025.
- [51] J. Hejna et al., “Robot Data Curation with Mutual Information Estimators,” in *Robotics: Science and Systems (RSS)*, 2025.
- [52] C. Agia et al., “CUPID: Curating Data your Robot Loves with Influence Functions,” in *Conference on Robot Learning (CoRL)*, 2025.
- [53] R. N. Landers, K. N. Bauer, and R. C. Callan, “Gamification of Task Performance with Leaderboards: A Goal Setting Experiment,” *Computers in Human Behavior*, 2017.
- [54] R. Farzan, J. M. DiMicco, D. R. Millen, B. Brownholtz, W. Geyer, and C. Dugan, “When the Experiment is Over: Deploying an Incentive System to All the Users,” in *Symposium on Persuasive Technology*, 2008.
- [55] C. Lynch et al., “Learning Latent Plans from Play,” in *Conference on Robot Learning (CoRL)*, 2020.
- [56] Z. J. Cui, Y. Wang, N. M. M. Shafiullah, and L. Pinto, “From Play to Policy: Conditional Behavior Generation from Uncurated Robot Data,” *International Conference on Learning Representations (ICLR)*, 2023.
- [57] Yixin Lin, Austin S. Wang, Giovanni Sutanto, Akshara Rai, and Franziska Meier, *Polymetis*, [facebookresearch.github.io/fairo/polymetis](https://facebookresearch.github.io/fairo/polymetis).

## APPENDIX

### A. System Implementation Details

We implement RoboCode with the Franka FR3 robot arm. Similar to the DROID setup [12], we mount two third-person ZED2 cameras and one ZED Mini camera on the robot's wrist. We calibrate the external cameras and merge point clouds via point-to-point ICP registration. Compressed point clouds are sent to the frontend and displayed alongside a virtual rendering of the robot via Three.js [45]. The camera angle of the 3D viewer moves based on the position of the end-effector to provide the user with a sense of depth, and a virtual beam projects the direction of the end-effector onto the table. We use a publisher-subscriber messaging pattern with ZeroMQ [43] for data processing on the backend, and WebSockets to communicate between the backend and the frontend. The frontend is implemented with Next.js. User accounts and authentication are managed using Supabase. As the input device, we use the Franka FR3 version of the open-source GELLO controller [16]. For safety, we perform collision avoidance with the table and workspace boundaries and additionally limit the magnitude of actions during remote teleoperation. We use joint impedance control via Polymetis [57]. We provide screenshots of the web interface in Fig. 7.

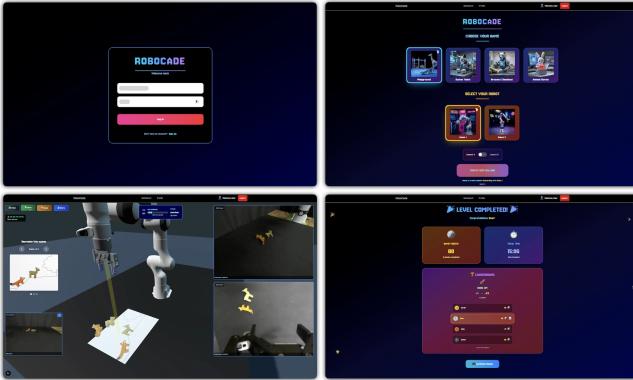


Fig. 7: **Web Interface.** (Top Left) Sign-in page. (Top Right) Task selection: Users select from available tasks and robots for remote teleoperation. (Bottom Left) Task play: While remotely controlling the real robot, users can view a rendering of the robot and objects in the scene. (Bottom Right) Post-task: Badges and leaderboard are presented post-task.

### B. Target Task Details

**ArrangeDesk.** This task involves picking up two pieces of electronics (a USB adapter and a computer mouse) placed in an arbitrary configuration on one side of a desk, alongside a mug, and arranging them in a straight line on the other side of the desk. The order, position, and orientation of the mug, adapter, and mouse are randomized in each episode. Picking the first object, placing the first object, picking the second object, and placing the second object in the proper positions constitute the 4 subtasks used to score this task.

**ScanBottle.** This task involves picking up a bottle of threadlocker solution and orienting the barcode in front of a scanning camera. The position and orientation of the bottle are randomized within a roughly 1.5-square-foot region. The

following subtasks are used to score this task: reaching the bottle such that the gripper makes contact with the bottle, picking up the bottle from the table, and scanning the bottle such that the correct side of the bottle is at least partly visible by the scanning camera.

**PackBox.** This task involves picking up a roll of electrical tape, placing it inside a cardboard box, and closing the box shut. The position of the box and tape are both randomized. The box lid cannot fully close if the tape roll is placed against an edge of the box. The following subtasks are used to score this task: picking up the tape from the table, inserting it into the box, and closing the lid of the box shut.

### C. Policy Learning Details

In this section, we provide hyperparameters for our policy training experiments. Hyperparameters for Diffusion Policy training are given in Table I and for VLA fine-tuning are given in Table II. For fair comparison, we use a fixed number of training steps for training and co-training policies (300K for Diffusion Policy). For  $\pi_{0.5}$  fine-tuning, we perform model selection among 5K, 10K, and 20K training steps with real-world evaluation on the target-only fine-tuned policy and use the same number of training steps for co-fine-tuning.

Training Steps	300K
Batch Size	128
Optimizer	AdamW
$\beta_1$	0.95
$\beta_2$	0.999
Learning Rate	1e-4
Weight Decay	1e-6
LR Schedule	Cosine Decay
Warmup Steps	2000
Diffusion Architecture	ConvID UNet
Prediction Horizon	16
Observation History	1
Num Action	8
Step Embedding Dim	128
UNet Down Dims	[256, 512, 1024]
Num Diffusion Steps	100
Num Inference Steps	20
Inference Scheduler	DDIM
Image Embedding Dim	256
Cameras	Third Person, Wrist

TABLE I: Hyperparameters for Diffusion Policy

Base Model	$\pi_{0.5}$ -DROID
Batch Size	32
Optimizer	AdamW
$\beta_1$	0.9
$\beta_2$	0.95
Learning Rate	2.5e-5
Gradient Clip Norm	1.0
LR Schedule	Cosine Decay
Warmup Steps	1000
Prediction Horizon	16
Num Action	10
Cameras	Third Person, Wrist

TABLE II: Hyperparameters for  $\pi_{0.5}$  Fine-tuning