

RoboCopilot: Human-in-the-loop Interactive Imitation Learning for Robot Manipulation

Philipp Wu^{*,†} Yide Shentu^{*,†} Qiayuan Liao[†] Ding Jin Menglong Guo

Koushil Sreenath[†] Xingyu Lin^{†,‡} Pieter Abbeel^{†,‡}

^{*}Equal contribution, [†] Equal advising
[†]University of California, Berkeley



Figure 1: A depiction of our RoboCopilot System which consists of a 20 degrees of freedom mobile bimanual robot and a bilateral teleoperation device. Our system enables easy teleoperation as well as human take-over at any time, allowing for an effective human-in-the-loop teleoperation system for interactive learning.

Abstract: Learning from human demonstration is an effective approach for learning complex manipulation skills. However, existing approaches heavily focus on learning from passive human demonstration data for its simplicity in data collection. Interactive human teaching has appealing theoretical and practical properties, but they are not well supported by existing human-robot interfaces. This paper proposes a novel system that enables seamless control switching between human and an autonomous policy for bi-manual manipulation tasks, enabling more efficient learning of new tasks. This is achieved through a compliant, bilateral teleoperation system. Through simulation and hardware experiments, we demonstrate the value of our system in an interactive human teaching for learning complex bi-manual manipulation skills.

Keywords: Learning from Human Demonstration; Robot Manipulation

1 Introduction

Humans play a critical role in teaching robot skills in robot manipulation. Recent developments in data-driven methods have shown promising results in learning complex robotic manipulation skills from human demonstration [1, 2, 3, 4, 5, 6]. The advantages of learning-from-demonstration approaches arise partly from their simplicity: humans provide a dataset of demonstration trajectories and then a policy is trained on the collected dataset to mimic the human actions. With enough human demonstrated data the policy will eventually learn to perform the same task like the human demonstrator.

However, this passive imitation learning paradigm suffers from various inefficiencies that an interactive learning approach could overcome. In an interactive learning setting, humans take control of the robot when it fails and provide demonstrations for corrective behaviors. Prior works in Dataset Aggregation (DAgger) have shown, both theoretically and empirically, that corrective and interactive data improve policy learning performance by addressing the covariate shift issue [7, 8, 9]. However, existing human teleoperation systems for manipulation are not designed for human intervention. Remote controllers such as spacemice and VR devices map the relative pose change to the robot end-effector, which is not intuitive for users, especially in an interactive setting where the human may need to take over at any time [6, 3]. Conversely, while exoskeleton-based devices have been effective in collecting passive human data [1, 3, 10], synchronizing the policy rollout to an exoskeleton can be potentially unsafe for the human operator in a DAgger setting [11]. Despite the recent advances of the easy-to-use teleoperation tools for multi-joints robot systems [3, 1], the challenge of enabling intuitive interactive learning with these tools remains unresolved.

In this paper, we introduce RoboCopilot, a robot system designed to accommodate **interactive human demonstrations** for learning bimanual robotic manipulation skills. We name our system RoboCopilot as it enables a human and a robot to cooperatively perform manipulation tasks, where the robot (the copilot) takes an assistive role before a lot of training. The system has two key components. First, we adopt the algorithmic framework introduced by Human-Gated DAgger (HG-DAgger) for interactive imitation learning and adapt it to our manipulation setting. In this pipeline, we alternate between model training and data collection with a learned policy, where, during data collection, a human teleoperator interrupts and provides corrective feedback. This enables the robot to continually improve performance throughout the teaching process and collect higher-quality interactive data. We first verify the benefit of this procedure through controlled simulation experiments. Second, we build custom hardware which consists of a mobile, compliant bimanual manipulator and a bilateral teleoperation device which enables us to instantiate our interactive learning pipeline. Our system enables a teleoperator to **seamlessly take over robot control** from a policy during the interactive training process. We show that with RoboCopilot, we can successfully teach the policy to learn long horizon, contact-rich, bimanual mobile manipulation skills.

2 Related Work

2.1 Imitation Learning from Passive Human Demonstrations

Human demonstrations provide high-quality, action-labeled data for learning complex manipulation skills. Prior works show that behavior cloning is effective for learning from passive human data [1, 2, 3, 4, 12, 13], and increasing the number of trajectories consistently leads to better performance. However, it is challenging to provide performance guarantees for the trained policies. The number of demonstrations required is task-dependent, ranging from as few as ten demonstration trajectories for a simple picking skill [14] to up to a thousand trajectories [15]. Furthermore, due to the issue of covariate shift, policies trained from passive offline data are unable to recover from the errors accumulated during online execution [7, 16]. Consequently, learning robust policies often has a significant hidden cost of an iterative process of data collection and policy learning. Our proposed system aims to enable interactive human teaching of complex manipulation skills, allowing the policy to continuously improve from newly collected demonstration data.

2.2 Interactive Imitation Learning

In an interactive imitation learning setting, the student policy receives feedback from the expert policy during policy execution, enabling the student to continually improve and learn to correct rollout mistakes [17]. Interactive learning approaches such as Data Aggregation (DAgger) have shown better sample efficiency and robustness compared to behavior cloning methods, both theoretically and in practice [7, 8]. However, retroactively relabeling demonstrations, as originally formulated, can be quite difficult for a human, especially for robot manipulation [18, 19]. Followup works have extended the DAgger framework to various settings to perform human-in-the-loop data collec-

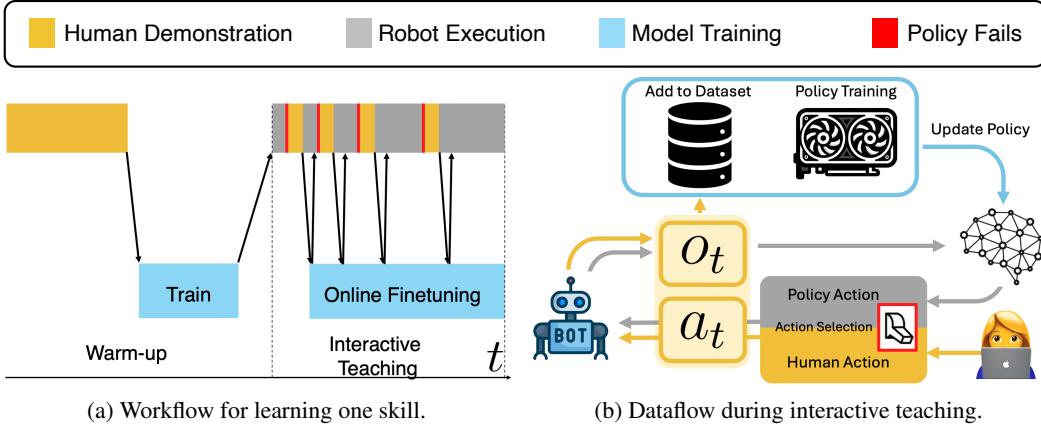


Figure 2: Overview for our interactive teaching system. (a) Workflow for learning a single skill: We start with a set of human demonstrations to pre-train the initial policy. Then in the interactive teaching stage, the policy is deployed and a human intervenes upon policy failure. The policy is continually fine-tuned from these new demos. As policy performance improves, less human intervention is needed. (b) During robot execution, the robot policy takes sensor observations and outputs actions. The human can decide when to use the policy switch to teleoperation. This enables the human to interrupt the robot on policy failure and correct the mistake, storing the data into the dataset. The model is continually training and being updated.

tion [11, 20, 21, 22, 23, 24, 25]. We follow Human-Gated DAgger (HG-DAgger) [11] on its data collection phase, where the human operator determines when to intervene and take control of the student policy and when to release control back to the policy. However, such an approach requires the human and the agent to switch control when needed, which can be challenging for a typical teleoperation device. As such, the DAgger framework has not been widely adopted for complex bi-manual manipulation. RoboCopilot aims to bridge this gap through a teleoperation system that enables seamless human take over and continual policy improvement.

2.3 Robot Teleoperation Systems

Teleoperation systems have existed from the very beginning of robotics [26] and the design for the teleoperation interface heavily impacts the final performance for modern data driven methods. Controllers such as joysticks and VR controllers [6, 27, 28, 29], while simple to set up, do not allow force feedback from the robot to the human operators and hide the kinematic joint limits of the robot from the human operators. Similarly, portable data collectors such as Universal Manipulation Interface (UMI) [30, 31, 32, 33] also pose a gap by removing the kinematic constraints from the data collection process. While exoskeleton-based puppeteering solutions have shown impressive results in collecting and learning fine-grained bimanual manipulation policies [1, 3, 34, 10], the controllers lose synchronization during policy execution, making it difficult for human to intervene. Most related to ours, there has been a body of prior works on bilateral teleoperation [35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45]. In this work, we build a compliant, bimanual teleoperation system and demonstrate its capability for interactive human teaching.

3 Continual Interactive Imitation Learning

We aim to enable robots to continually learn from interactive demonstrations, where a human can take over control of the robot when the autonomous policy fails, as illustrated in Figure 2. As human demonstration, robot execution and skill learning occur in a tight learning loop, this allows the human operator to understand where and when the autonomous policy fails, providing more targeted demonstrations on the model failure cases. We will summarize our interactive learning algorithm in this section and explain how we achieve this through our proposed RoboCopilot system in the next section.

As shown in Algorithm 1, we first collect K human demonstration trajectories to warm up the policy. Following HG-DAGger [22], we allow the human expert to determine when to intervene and take control of the robot execution. The intervention data from the human experts are added to the dataset. After each round of data collection, we fine-tune our policy using all the collected data by performing several gradient-based updates to the policy network. Our policy network is parameterized by a diffusion network [2]. As the policy improves, the number of required interventions decreases significantly.

Prior work has shown that continually fine-tuning neural networks on data from non-stationary distributions causes catastrophic forgetting [46, 47]. Despite this, we adopt fine-tuning during the interactive teaching process give its multiple advantages over training policies from scratch after the data collection is done: First, collecting online samples from the current policy alleviates covariate shift, as the human can intervene when necessary to teach correction skills. Second, as the policy is continually fine-tuned and executed, the human operator can quickly observe current failure modes and collect more targeted demonstrations. Finally, online learning reduces the overall training time.

Practically, once the interactive teaching finishes, we can utilize the online interactive data and retraining a policy from scratch. Doing so avoids the non-stationarity of the training distribution but still takes advantage of the interactive data. In practice, we find this approach to provide the best policy that can be used for deployment. This amounts to restarting Algorithm 1 with the most up to date dataset D .

4 Teleoperation System

We develop a versatile mobile bi-manual robot with a capable teleoperation system that allows us to instantiate human-in-the-loop interactive teaching for a wide range of real world tasks. Our physical RoboCopilot teleoperation system consists of the physical robot, the teleoperation device, and the teleoperation workflow. More details about our custom low cost robot hardware can be found in Appendix D.

Our teleoperation system is an effective way to enable human-in-the-loop imitation learning for manipulation. A more capable and human intuitive device enables a user to more precisely control the robot which offers increased data quality for downstream learning. While for dynamical systems with a lower dimensional action space like UAVs or cars, one can more easily apply DAGger algorithms by having a human relabel the data with a joystick or wheel [48, 11]. In general, there is no obvious way to relabel trajectory

Algorithm 1 Human-in-the-loop Imitation Learning

```

1: Input: Init policy  $\pi_0$ , human expert  $\pi^*$ , # of iterations  $N$ ,  
warmup human demonstration dataset  $D$   

2: Train warmup policy  $\pi_1$  on  $D$  with behavior cloning  

3: for  $i = 2$  to  $N$  do  

4:   for each step in the environment do  

5:     if human expert  $\pi^*$  intervenes then  

6:       Execute expert human action:  $\pi^*(s)$   

7:       Add the intervention data:  $D \leftarrow D \cup \{(s, \pi^*(s))\}$   

8:     else  

9:       Execute robot policy  $\pi_i(s)$   

10:    end if  

11:   end for  

12:    $\pi_{i+1}(s) \leftarrow \text{finetune}(\pi_i, D)$   

13: end for  

14: Output: Final policy  $\pi_N$ 

```

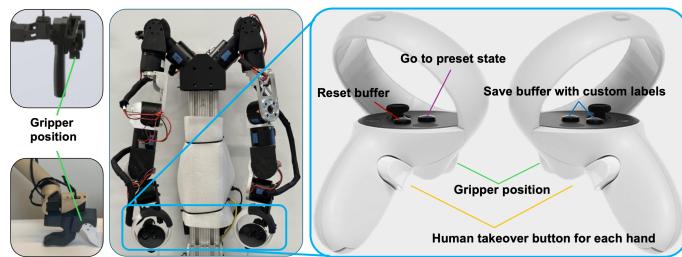


Figure 3: The comparison of the different end-effector human interface designs for different teleoperation systems is shown below: Left: Aloha [1] and GELLO [3]’s handheld interface. Middle: RoboCopilot layout, where we attached the Quest2 controller at the end of our GELLO device. Right: The key map of our end-effector human input interface. We optimized the layout to allow efficient gripper control and interactive human-in-the-loop teaching.

ries for manipulation trajectory data. This is due in part to the high dimensional action space and complex environment dynamics.

4.1 Teleoperation Device

The core principle of the teleoperation leader device is to enable maximal data collection quality and ease of use for the human during operation and take over. Our puppeteering teleoperation device is a low cost approximate kinematic replica of our target arm, following GELLO [3]. More detailed hardware specifications are found in Appendix D. We attach a Meta Quest 2 controller [49] at the end of our teleoperation leader arms, serving as both an ergonomic gripping handle and a multi-input control device. Previous works have primarily focused on the active portion of controlling the arm, but our approach also considers the importance of a comprehensive and user-friendly interface. A comparison of different teleoperation devices is shown in Figure 3. While teaching devices like ours are not new, the focus has largely been in an imitation learning context. Our goal is to show how to effectively extend the usefulness of these devices in an interactive learning setting.

4.2 Active Control

We actively control the motors of the leader arm enabling less user fatigue and smoother human take over. The weight and inertia of the teleoperation device requires additional energy input from the user to drive the device. Active gravity compensation using the manipulator equation alleviates this burden, allowing the operator to maneuver the system without excessive effort, reducing operator fatigue [50, 51].

Active control also enables a bilateral system, where the leader arm can provide force feedback to the user felt by the follower arm. This enables a user to understand and feel the forces that the robot feels, allowing the user to more effectively operate the robot. We modify the PD control law used in Katz [36] control law so the user feels a scaled down version of the forces [36].

$$\tau_L = \alpha K_p(\theta_F - \theta_L) + \beta K_d(\dot{\theta}_F - \dot{\theta}_L) \quad (1)$$

$$\tau_F = K_p(\theta_L - \theta_F) + K_d(\dot{\theta}_L - \dot{\theta}_F) \quad (2)$$

where L is the leader teleoperation device and F is the follower, τ is actuator torques, θ is joint position and $\dot{\theta}$ is joint velocity. We add α and β constants so that the teleoperation device does not feel the full inertia of the arm, while still enabling effective control. Additionally, the use of active motors on our teleoperation device allows us to flexibly vary the α, β bilateral coefficient in different situations.

4.3 Human-in-the-loop Interactive Learning

Our RoboCopilot system is designed to be user-friendly and fully equipped with features that facilitate both the smooth operation of the robot for task completion and efficient data collection. The teleoperation device includes multiple buttons dedicated to data collection utilities, such as saving data and resetting the robot, streamlining the process of capturing valuable data during operation. This enables the teleoperator to be explicit about the data being collected, and can easily control for the data quality.

Control of the leader arm is intuitive, with bilateral control allowing the user to feel the forces experienced by the robot arm. This feature is particularly beneficial when the robot is in contact with the environment or handling heavy objects. When the autonomous policy is executing, the leader-follower relationship is reversed where the teleoperation device matches the real robot. The teleoperation can then activate take over at any point in time by engaging with the teleoperation device, without interrupting process flow as the devices are already synchronized. During robot policy execution, we use stiff gains on the leader to minimize the state difference, whereas during teleoperation mode, we use lower gains to ease the forces on the human teleoperator.

Table 1: Simulation results comparing behavior cloning and various DAgger methods on the robomimic benchmark [14]. We collect 10 trajectories for warming up the dataset and training the initial policy. BC-H refers to behavior cloning from a human demonstration dataset. BC-P refers to behavior cloning from an expert dataset collected by our expect policy. The expert policy is the same policy that makes interventions in our DAgger methods. **Continual DAgger** denotes our continual DAgger method that continually finetuning with interaction data. **Batched DAgger** denotes training from scratch with the Continual DAgger collected data. We see leveraging human in the loop during the data collection process significantly improves over the baselines. For our DAgger runs, we initialize with 10 expert demo trajectories. Experiments are run across three seeds.

Environment	Total Number of Trajectories	BC-H	BC-P	Continual DAgger (Ours)	Batched DAgger (Ours)
Can	10 (warmup)	0.28±0.04	0.43±0.0	—	—
Can	15	0.45±0.02	0.57±0.04	0.57±0.02	0.63±0.02
Can	20	0.61±0.04	0.62±0.02	0.62±0.02	0.72±0.02
Can	30	0.76±0.01	0.82±0.02	0.85±0.04	0.83±0.02
Can	40	0.83±0.03	0.93±0.01	0.87±0.02	0.93±0.02
Square	10 (warmup)	0.26±0.02	0.30±0.02	—	—
Square	15	0.38±0.05	0.28±0.03	0.39±0.02	0.43±0.02
Square	20	0.40±0.00	0.41±0.03	0.43±0.01	0.44±0.03
Square	30	0.48±0.05	0.46±0.03	0.47±0.02	0.55±0.02
Square	40	0.52±0.01	0.51±0.02	0.53±0.05	0.57±0.01
Transport	10 (warmup)	0.41±0.10	0.65±0.04	—	—
Transport	15	0.55±0.02	0.57±0.05	0.72±0.06	0.87±0.02
Transport	20	0.53±0.05	0.70±0.02	0.88±0.02	0.97±0.02
Transport	30	0.65±0.04	0.61±0.08	0.90±0.04	0.92±0.02
Transport	40	0.77±0.04	0.83±0.01	0.90±0.04	0.93±0.02

5 Experiments

We conducted a series of experiments to demonstrate the value of our system for continual, interactive imitation learning in both simulation and the real world. Our system enables interactive data collection with a human in the loop. Our primary research question is how this interactive data collection affects the collected data quality. We measure the quality of the collected data by evaluating the final policy performance. Specifically, we compare the policy performance using three data collection and policy learning methods: (1) Offline Behavior Cloning (BC), (2) Human-in-the-loop interactive learning where the policy is continually trained and used for collecting the next batch of data (Continual DAgger), and (3) Training a policy from scratch using all data, including the data collected by Continual DAgger (Batched Dagger). Through these experiments, we aim to demonstrate that interactive data collection can accelerate learning, reduce the need for laborious data collection process, and improve the overall performance. Additionally we demonstrate the effectiveness of our real world system at learning various tasks.

5.1 Simulation

We conduct imitation learning experiments on the standard Robomimic benchmark [14]. The benchmark provides human demonstration trajectories and simulation environments for a set of robotic manipulation tasks including bimanual tasks, showing the benefit of our approach for manipulation in general. The goal of these experiments is to validate the effectiveness of the algorithmic approach in a controlled reproducible environment. We choose the Can, Square, and Transport tasks from the benchmark which are visualized in Appendix B. Experiments involving human interaction can be difficult to reproduce due to the volatile nature of human interaction. To improve reproducibility, we use the pre-trained expert diffusion policy π^* from Chi et al. [2] as a human substitute. As this trained expert cannot decide when to intervene, we follow Algorithm 1 but use $\pi = (1 - \beta)\pi_i + \beta\pi^*$ as our rollout policy when $|\pi_i(s_t) - \pi^*(s_t)| > \epsilon$. Here, π_i is our training policy at the i^{th} iteration.

The results are shown in Table 1. Our DAgger variants use 10 expert demos to initialize, with the rest being DAgger-corrected trajectories. Our continual learning method achieves superior performance to behavior cloning baselines, despite the issue of catastrophic forgetting. Some benefits of continual

Table 2: Task completion rate of the industrial picking experiment we ran in the real environment. From this table, we can see that under the human-in-the-loop interactive learning setting, the policy can achieve higher performance given the same amount of human-labeled trajectories. With the human-in-the-loop copilot data collection process, the learned policy achieves a much higher success rate given the same number of training trajectories. Additionally, since the human operator can allow the policy to take over again once a mistake is corrected, the number of human-teleoperated timesteps in these DAggered trajectories is significantly less compared to those in offline BC data.

Task	# of Trajectories	Offline BC	Continual DAgger (Ours)	Batched DAgger (Ours)
Industrial Picking	12 (warmup)	38.9%	—	—
	24	61.1%	72.2%	72.2%
	36	66.7%	72.2%	77.7%
Industrial Part Transport	10 (warmup)	33.3%	—	—
	15	50.0%	66.7%	75.0%
	20	58.3%	75.0%	91.7%

learning are hard to show with a learned policy trained on the distribution of human demos, as the policy does not have the ability to give sophisticated corrective demonstration nor intelligently collect targeted demonstrations in areas of failure. Even so, our simulation results demonstrate the value of interactive data, as seen by the performance boost provided by Continual DAgger. This increase in data quality is further supported by the improvement of the policy in our Batched DAgger training where we train the policy from scratch using the Continual DAgger data. As expected, Batched DAgger greatly outperforms the continually trained variant in almost all cases. This greatly motivates the advantages of DAgger like training for manipulation. Next we test the effectiveness of our RoboCopilot system in real world settings using a human as the expert demonstrator, leveraging our teleoperation device to interrupt and interact with the scene.

5.2 Real World

Our real-world experiments provide similar conclusions to our simulation experiments, where interactive data collection provides higher quality data than passive data collection from the expert policy. We consistently observe that the success rate keeps increasing as we collect more interactive data, and the need for human intervention decreases accordingly. The need to intervene also serves as a useful signal in practice for when to stop collecting new data.

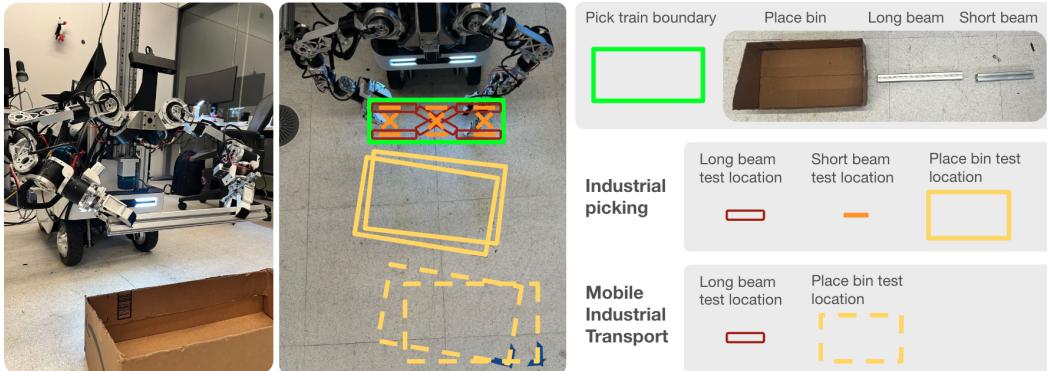


Figure 4: An illustration of the evaluation protocol for the industrial part transport tasks. During training, the beam is placed in different positions within a defined boundary (highlighted in green). Industrial picking requires the robot to locate and manipulate the long beam or short beam and place it within the bin. Mobile industrial picking only considers the long beam, but the bin is further away, requiring the robot to drive the base before placing. We label the poses of the beams and the bin to ensure consistency during evaluation.

5.2.1 Part Picking

We test our RoboCopilot system in the real world on an industrial part picking task. The goal of the robot in this task is to pick up industrial aluminum extrusions and place them into a bin. We

use two different sized aluminum extrusions, a small one which can be picked up with a single arm, a longer beam should be picked up by both arms. To control the experiment we define a domain of interest where the 8020 beams are placed. Additionally, for the placing bin, we mark 3 train locations, as well as 2 test locations. During testing of a policy, we choose and evaluate the robot of 18 predetermined locations. This task allows us to study various components of our system. First, the setting is very multimodal, as there may be different strategies and methods of how to pick up the object. Second we can test the effectiveness compliance of our system as holding the long beam with both hands is more straightforward for a compliant system.

We define four “task types”: large bar, small bar left hand, small bar right hand, and small bar middle both hands. The primary question we aim to address is how effective our shared autonomy system is for collecting demonstrations to learn new skills. To start the experiment, we collect 12 trajectories (3 for each subtask) to initialize the policy, ensuring that it does not behave randomly or completely ineffectively. During the policy rollout, the human teleoperator has the option to take over from the autonomous system if the policy fails to complete the task. For each human-in-the-loop DAgger process, we will continue running the autonomous system until the human operator takes over, three times for each subtask. We tested our continuous learning pipeline under two different settings: Batched DAgger and Continual DAgger. For the Batched DAgger setting, every time we collect new interactive data, we relaunch the training process from scratch. In contrast, for the Continual DAgger setting, we continuously stream the human interactive data into the training pipeline to continuously fine-tune the model. All DAgger runs use policies from the warmup offline BC checkpoint.

5.2.2 Part Transport

In addition to the stationary industrial part pick-and-place task, we applied the RoboCopilot pipeline to a more challenging industrial part transport task, which requires moving the robot base while performing manipulation tasks, as shown in Figure 6. Specifically for this task, the robot must navigate to the tote, which is placed far from its starting position, to successfully complete the placing task. See Appendix A for a visual depiction the task. In Table 2, we compare the performances of three methods given the same demonstration budget: Offline BC, which trains a policy from passively collected data; Continual DAgger, which continually fine-tunes a policy after collecting 10, 15, and 20 trajectories; and Batched DAgger, which uses the final dataset obtained by running Continual DAgger and trains a policy from scratch. During Continual DAgger, we observe that humans can identify initial configurations where the current policy fails and reset intelligently. In this way, we find that Continual DAgger is more data-efficient than Offline BC, thanks to intelligent reset. However, Continual DAgger keeps fine-tuning the same policy as new demonstrations are added to the dataset, leading to suboptimal policy optimization. In Batched DAgger, we train one policy from scratch using this interactively collected dataset and achieve the best performance, further validating our claim that human-in-the-loop data collection improves the quality of the collected data.

5.2.3 Kitchen

Lastly we study a long horizon task in a toy kitchen, requiring the robot to execute multiple individual steps in order to “cook the tomato”. This high level task is broken up into 4 smaller subtasks, where the robot first must open the cabinet door, then pick the tomato, then put the tomato in the pot, and lastly turn the dial. If the robot fails to accomplish any one task, we count failure for all the following steps as in the CALVIN long horizon benchmark [52]. See Fig 5 for an illustration of this task. We compare naive Offline BC to our interactive approach leveraging Batched DAgger. The results are found in Table 3. We find that leveraging human-in-the-loop not only results in better long horizon performance, but it also decreases the necessary human teleoperated time to reach that performance. Interestingly we find that more demos for Offline BC does not help the policy learn to fully complete the task, whereas our Batched DAgger approach enables a significant success increase in fully completing the sequence of subtasks, despite having a smaller overall dataset size. Our experiments show that Batched DAgger with one iteration of data collection and a total of 30 trajectories is able to match or outperform the success rates of Offline BC with 45 trajectories on all

Table 3: Experimental results on the long horizon kitchen task. We compare Offline BC with Batched DAgger for different trajectory dataset sizes, in increments of 15. The size of the training dataset is shown. For each experiment, we also show the number steps that required a human to operate the robot to collect the data at each stage. We also show a detailed breakdown of the task success rate for each subtask in the overall task.

# of Trajectories	Method	Dataset Size (# Transitions)	% intervention steps	Open Door	Grab Tomato	Place Tomato	Turn Dial
15	Offline BC	8491	100%	100%	70%	25%	5%
30	Offline BC	16654	100%	100%	70%	35%	5%
	Batched DAgger	15162	65 %	100%	85%	45%	20%
45	Offline BC	25694	100%	100%	85%	40%	0%
	Batched DAgger	20844	52%	100%	90%	50%	30%

subtasks, with the performance gap being the largest for completing the full task (i.e. Turn Dial). This shows that our approach is also effective in learning trajectories that have long time horizons. Lastly we see that Batched DAgger requires less human collected trajectory steps for a fixed number of training trajectories, as the human only needs to operate after a policy failure. This reduction in human intervention steps further underscores the efficiency of our human-in-the-loop system in teaching the robot

6 Discussion

Limitations. While our RoboCopilot system shows promising results in enabling interactive teaching and continual learning for bimanual robotic manipulation, it is not without limitations. First, the requirement for human intervention during the learning process can be resource-intensive, as it necessitates a skilled operator to oversee and correct the robot’s actions continually. This dependency might limit the scalability of our system in environments where such expertise is not readily available or the task is too difficult to even teleoperate. Second, although our system aims to be cost-efficient, the initial setup and maintenance costs might still be prohibitive for smaller organizations or research labs due to the large number of custom components. Additionally, the use of planetary gearboxes, while beneficial for compliance and cost, introduces some degree of backlash, which may affect the precision of fine manipulation tasks.



Figure 5: An overall illustration of the toy kitchen task. (Subtask 1) The robot needs to first open the spring-loaded cabinet door and hold the door open. (Subtask 2) The robot can pick the tomato and transfer it to the stove area. (Subtask 3): At this stage, the robot needs to put the tomato into the pot. Notice that the pot can be at different locations. (Subtask 4) Finally, the robot needs to turn the correct dial depending on which stove the pot is at.

Conclusions In this work, we introduced RoboCopilot, a novel system that enables human-in-the-loop interactive imitation learning for bimanual robotic manipulation tasks. The RoboCopilot system underscores the potential of seamlessly interfacing a human operator and an autonomous policy for interactive teaching. Future work will focus on enhancing its scalability and ease of deployment in diverse real-world environments.

Acknowledgments

.Pieter Abbeel holds concurrent appointments as a Professor at UC Berkeley and as an Amazon Scholar. This paper describes work performed at UC Berkeley and is not associated with Amazon.

References

- [1] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware, 2023.
- [2] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. *arXiv preprint arXiv:2303.04137*, 2023.
- [3] P. Wu, Y. Shentu, Z. Yi, X. Lin, and P. Abbeel. Gello: A general, low-cost, and intuitive teleoperation framework for robot manipulators, 2023.
- [4] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, J. Ibarz, B. Ichter, A. Irpan, T. Jackson, S. Jesmonth, N. J. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, K.-H. Lee, S. Levine, Y. Lu, U. Malla, D. Manjunath, I. Mordatch, O. Nachum, C. Parada, J. Peralta, E. Perez, K. Pertsch, J. Quiambao, K. Rao, M. Ryoo, G. Salazar, P. Sanketi, K. Sayed, J. Singh, S. Sontakke, A. Stone, C. Tan, H. Tran, V. Vanhoucke, S. Vega, Q. Vuong, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, and B. Zitkovich. Rt-1: Robotics transformer for real-world control at scale, 2022.
- [5] E. Jang, A. Irpan, M. Khansari, D. Kappler, F. Ebert, C. Lynch, S. Levine, and C. Finn. Bc-z: Zero-shot task generalization with robotic imitation learning. In *Conference on Robot Learning*, pages 991–1002. PMLR, 2022.
- [6] T. Zhang, Z. McCarthy, O. Jow, D. Lee, X. Chen, K. Goldberg, and P. Abbeel. Deep imitation learning for complex manipulation tasks from virtual reality teleoperation. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5628–5635. IEEE, 2018.
- [7] S. Ross, G. Gordon, and D. Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 627–635. JMLR Workshop and Conference Proceedings, 2011.
- [8] D. A. Pomerleau. Alvinn: An autonomous land vehicle in a neural network. *Advances in neural information processing systems*, 1, 1988.
- [9] E. Jang, A. Irpan, M. Khansari, D. Kappler, F. Ebert, C. Lynch, S. Levine, and C. Finn. BC-z: Zero-shot task generalization with robotic imitation learning. In *5th Annual Conference on Robot Learning*, 2021.
- [10] H. Fang, H.-S. Fang, Y. Wang, J. Ren, J. Chen, R. Zhang, W. Wang, and C. Lu. Low-cost exoskeletons for learning whole-arm manipulation in the wild. *arXiv preprint arXiv:2309.14975*, 2023.
- [11] M. Kelly, C. Sidrane, K. Driggs-Campbell, and M. J. Kochenderfer. Hg-dagger: Interactive imitation learning with human experts, 2019.
- [12] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn, P. Florence, C. Fu, M. G. Arenas, K. Gopalakrishnan, K. Han, K. Hausman, A. Herzog, J. Hsu, B. Ichter, A. Irpan, N. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, L. Lee, T.-W. E. Lee, S. Levine, Y. Lu, H. Michalewski, I. Mordatch, K. Pertsch, K. Rao, K. Reymann, M. Ryoo, G. Salazar, P. Sanketi, P. Sermanet, J. Singh, A. Singh, R. Soricut, H. Tran, V. Vanhoucke, Q. Vuong, A. Wahid, S. Welker, P. Wohlhart, J. Wu, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, and B. Zitkovich. Rt-2: Vision-language-action models transfer web knowledge to robotic control, 2023.

- [13] S. Haldar, J. Pari, A. Rai, and L. Pinto. Teach a robot to fish: Versatile imitation from one minute of demonstrations. *arXiv preprint arXiv:2303.01497*, 2023.
- [14] A. Mandlekar, D. Xu, J. Wong, S. Nasiriany, C. Wang, R. Kulkarni, L. Fei-Fei, S. Savarese, Y. Zhu, and R. Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. *arXiv preprint arXiv:2108.03298*, 2021.
- [15] K. Bousmalis, G. Vezzani, D. Rao, C. Devin, A. X. Lee, M. Bauza, T. Davchev, Y. Zhou, A. Gupta, A. Raju, A. Laurens, C. Fantacci, V. Dalibard, M. Zambelli, M. Martins, R. Pevcović, M. Blokzijl, M. Denil, N. Batchelor, T. Lampe, E. Parisotto, K. Źołna, S. Reed, S. G. Colmenarejo, J. Scholz, A. Abdolmaleki, O. Groth, J.-B. Regli, O. Sushkov, T. Rothörl, J. E. Chen, Y. Aytar, D. Barker, J. Ortiz, M. Riedmiller, J. T. Springenberg, R. Hadsell, F. Nori, and N. Heess. Robocat: A self-improving foundation agent for robotic manipulation, 2023.
- [16] J. Spencer, S. Choudhury, A. Venkatraman, B. Ziebart, and J. A. Bagnell. Feedback in imitation learning: The three regimes of covariate shift. *arXiv preprint arXiv:2102.02872*, 2021.
- [17] C. Celemin, R. Pérez-Dattari, E. Chisari, G. Franzese, L. de Souza Rosa, R. Prakash, Z. Ajanović, M. Ferraz, A. Valada, J. Kober, et al. Interactive imitation learning in robotics: A survey. *Foundations and Trends® in Robotics*, 10(1-2):1–197, 2022.
- [18] M. Laskey, J. Lee, C. Chuck, D. Gealy, W. Hsieh, F. T. Pokorny, A. D. Dragan, and K. Goldberg. Robot grasping in clutter: Using a hierarchy of supervisors for learning from demonstrations. In *2016 IEEE International Conference on Automation Science and Engineering (CASE)*, pages 827–834, 2016. doi:10.1109/COASE.2016.7743488.
- [19] M. Laskey, J. Lee, R. Fox, A. Dragan, and K. Goldberg. Dart: Noise injection for robust imitation learning, 2017.
- [20] R. Hoque, A. Balakrishna, E. Novoseller, A. Wilcox, D. S. Brown, and K. Goldberg. Thriftydagger: Budget-aware novelty and risk gating for interactive imitation learning, 2021.
- [21] Q. Li, Z. Peng, and B. Zhou. Efficient learning of safe driving policy via human-ai copilot optimization, 2022.
- [22] A. Mandlekar, D. Xu, R. Martín-Martín, Y. Zhu, L. Fei-Fei, and S. Savarese. Human-in-the-loop imitation learning using remote teleoperation. *arXiv preprint arXiv:2012.06733*, 2020.
- [23] H. Liu, S. Nasiriany, L. Zhang, Z. Bao, and Y. Zhu. Robot learning on the job: Human-in-the-loop autonomy and learning during deployment, 2023.
- [24] J. Spencer, S. Choudhury, M. Barnes, M. Schmittie, M. Chiang, P. J. Ramadge, and S. S. Srinivasa. Learning from interventions: Human-robot interaction as both explicit and implicit feedback. *Robotics: Science and Systems XVI*, 2020.
- [25] J. Zhang and K. Cho. Query-efficient imitation learning for end-to-end autonomous driving, 2016. URL <https://arxiv.org/abs/1605.06450>.
- [26] S. Lichiardopol. A survey on teleoperation. 2007.
- [27] Y. Zhu, A. Joshi, P. Stone, and Y. Zhu. Viola: Imitation learning for vision-based manipulation with object proposal priors. *CoRL*, 2022.
- [28] S. Dass, W. Ai, Y. Jiang, S. Singh, J. Hu, R. Zhang, P. Stone, B. Abbate, and R. Martin-Martin. Telemoma: A modular and versatile teleoperation system for mobile manipulation. *arXiv preprint arXiv:2403.07869*, 2024.
- [29] Y. Qin, W. Yang, B. Huang, K. Van Wyk, H. Su, X. Wang, Y.-W. Chao, and D. Fox. Anyteleop: A general vision-based dexterous robot arm-hand teleoperation system. *RSS*, 2023.

- [30] S. Song, A. Zeng, J. Lee, and T. Funkhouser. Grasping in the wild: Learning 6dof closed-loop grasping from low-cost demonstrations. *IEEE Robotics and Automation Letters*, 5(3):4978–4985, 2020.
- [31] S. Young, D. Gandhi, S. Tulsiani, A. Gupta, P. Abbeel, and L. Pinto. Visual imitation made easy. *CoRL*, 2020.
- [32] C. Chi, Z. Xu, C. Pan, E. Cousineau, B. Burchfiel, S. Feng, R. Tedrake, and S. Song. Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots. *arXiv preprint arXiv:2402.10329*, 2024.
- [33] N. M. M. Shafiullah, A. Rai, H. Etukuru, Y. Liu, I. Misra, S. Chintala, and L. Pinto. On bringing robots home. *arXiv preprint arXiv:2311.16098*, 2023.
- [34] Z. Fu, T. Z. Zhao, and C. Finn. Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation. *arXiv preprint arXiv:2401.02117*, 2024.
- [35] T. Hulin, K. Hertkorn, P. Kremer, S. Schätzle, J. Artigas, M. Sagardia, F. Zacharias, and C. Preusche. The dlr bimanual haptic device with optimized workspace. In *2011 IEEE International Conference on Robotics and Automation*, pages 3441–3442. IEEE, 2011.
- [36] B. Katz. A low cost modular actuator for dynamic robots. *PhD Thesis*, 2018.
- [37] M. Schwarz, C. Lenz, A. Rochow, M. Schreiber, and S. Behnke. Nimbro avatar: Interactive immersive telepresence with force-feedback telemanipulation. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5312–5319. IEEE, 2021.
- [38] J. Elsner, G. Reinerth, L. Figueiredo, A. Naceri, U. Walter, and S. Haddadin. Parti-a haptic virtual reality control station for model-mediated robotic applications. *Frontiers in Virtual Reality*, 3:925794, 2022.
- [39] C. Lenz and S. Behnke. Bimanual telemanipulation with force and haptic feedback and predictive limit avoidance. *CoRR*, abs/2109.13382, 2021.
- [40] E. Ackerman. Beautiful fluid actuators from disney research make soft, safe robot arms, 2014.
- [41] Y. Ishiguro, T. Makabe, Y. Nagamatsu, Y. Kojio, K. Kojima, F. Sugai, Y. Kakiuchi, K. Okada, and M. Inaba. Bilateral humanoid teleoperation system using whole-body exoskeleton cockpit tablis, 08 2020.
- [42] S. Tachi, K. Komoriya, K. Sawada, T. Nishiyama, T. Itoko, M. Kobayashi, and K. Inoue. Telexistence cockpit for humanoid robot control, 2003.
- [43] A. Amini, A. Mirbagheri, and A. Homayoun Jafari. Bilateral control of a nonlinear teleoperation robotic system with time varying delay using optimal control method. In *2017 24th National and 2nd International Iranian Conference on Biomedical Engineering (ICBME)*, pages 330–333, 2017.
- [44] T. Buamanee, M. Kobayashi, Y. Uranishi, and H. Takemura. Bi-act: Bilateral control-based imitation learning via action chunking with transformer, 2024.
- [45] M. Mohsen, A. M. Mohamed, S. Ahmed, and K. Ibrahim. Bilateral control of a 2-dof teleoperated manipulator using udp scheme. *Ain Shams Engineering Journal*, 14(9):102065, 2023. ISSN 2090-4479.
- [46] I. J. Goodfellow, M. Mirza, D. Xiao, A. Courville, and Y. Bengio. An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint arXiv:1312.6211*, 2013.

- [47] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.
- [48] S. Ross, N. Melik-Barkhudarov, K. S. Shankar, A. Wendel, D. Dey, J. A. Bagnell, and M. Hebert. Learning monocular reactive uav control in cluttered natural environments, 2012.
- [49] Meta. Meta quest 2. <https://www.meta.com/quest/products/quest-2/>, 2024.
- [50] B. Siciliano and O. Khatib. *Springer Handbook of Robotics*. Springer-Verlag, Berlin, Heidelberg, 2007. ISBN 354023957X.
- [51] K. M. Lynch and F. C. Park. *Modern Robotics: Mechanics, Planning, and Control*. Cambridge University Press, USA, 1st edition, 2017. ISBN 1107156300.
- [52] O. Mees, L. Hermann, E. Rosete-Beas, and W. Burgard. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks, 2022. URL <https://arxiv.org/abs/2112.03227>.
- [53] Intel. Realsense d405. <https://www.intelrealsense.com/depth-camera-d405/>, 2024.
- [54] G. Pratt and M. Williamson. Series elastic actuators. In *2011 IEEE International Conference on Robotics and Automation*, 1995.
- [55] A. Albu-Schäffer, S. Haddadin, C. Ott, A. Stemmer, T. Wimböck, and G. Hirzinger. The dlr lightweight robot – design and control concepts for robots in human environments. *INDUSTRIAL ROBOT-AN INTERNATIONAL JOURNAL*, 34:376–385, 08 2007. doi:10.1108/01439910710774386.
- [56] K. A. Wyrobek, E. H. Berger, H. M. V. der Loos, and J. K. Salisbury. Towards a personal robotics development platform: Rationale and design of an intrinsically safe personal robot. *2008 IEEE International Conference on Robotics and Automation*, pages 2165–2170, 2008.
- [57] D. V. Gealy, S. McKinley, B. Yi, P. Wu, P. R. Downey, G. Balke, A. Zhao, M. Guo, R. Thomasson, A. Sinclair, P. Cuellar, Z. McCarthy, and P. Abbeel. Quasi-direct drive for low-cost compliant robotic manipulation, 2019.
- [58] M. Quigley, A. Asbeck, and A. Ng. A low-cost compliant 7-dof robotic manipulator. In *2011 IEEE International Conference on Robotics and Automation*, 2011.
- [59] B. Katz. A low cost modular actuator for dynamic robots. In <https://api.semanticscholar.org/CorpusID:69840472>, 2018.
- [60] V. Myers, A. He, K. Fang, H. Walke, P. Hansen-Estruch, C.-A. Cheng, M. Jalobeanu, A. Kolobov, A. Dragan, and S. Levine. Goal representations for instruction following: A semi-supervised language interface to control, 2023.
- [61] K. Fang, P. Yin, A. Nair, H. Walke, G. Yan, and S. Levine. Generalization with lossy affordances: Leveraging broad offline data for learning visuomotor tasks, 2023.
- [62] M. Guo, P. Wu, B. Yi, D. Gealy, S. McKinley, and P. Abbeel. Blue gripper: A robust, low-cost, and force-controlled robot hand. In *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*, pages 1505–1510, 2019. doi:10.1109/COASE.2019.8843134.
- [63] R. Ma and A. Dollar. Yale openhand project: Optimizing open-source hand designs for ease of fabrication and adoption. *IEEE Robotics and Automation Magazine*, 24(1):32–40, 2017. doi:10.1109/MRA.2016.2639034.

- [64] K. Shaw, A. Agarwal, and D. Pathak. Leap hand: Low-cost, efficient, and anthropomorphic hand for robot learning. *Robotics: Science and Systems (RSS)*, 2023.
- [65] K. Elgeneidy, P. Lightbody, S. Pearson, and G. Neumann. Characterising 3d-printed soft fin ray robotic fingers with layer jamming capability for delicate grasping. In *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*, pages 143–148, 2019. doi:10.1109/ROBOSOFT.2019.8722715.
- [66] W. Crooks, G. Vukasin, M. O’Sullivan, W. Messner, and C. Rogers. Fin ray® effect inspired soft robotic gripper: From the robosoft grand challenge toward optimization. *Frontiers in Robotics and AI*, 3, 2016. ISSN 2296-9144. doi:10.3389/frobt.2016.00070. URL <https://www.frontiersin.org/articles/10.3389/frobt.2016.00070>.
- [67] X. Shan and L. Birglen. Bistable Stopper Design and Force Prediction for Precision and Power Grasps of Soft Robotic Fingers for Industrial Manipulation. *Journal of Mechanical Design*, 146(4):045001, 11 2023. ISSN 1050-0472. doi:10.1115/1.4063763.
- [68] R. Holmberg and O. Khatib. Development and control of a holonomic mobile robot for mobile manipulation tasks. *The International Journal of Robotics Research*, 19(11):1066–1074, 2000. doi:10.1177/02783640022067977. URL <https://doi.org/10.1177/02783640022067977>.
- [69] J. Wu, R. Antonova, A. Kan, M. Lepert, A. Zeng, S. Song, J. Bohg, S. Rusinkiewicz, and T. Funkhouser. Tidybot: Personalized robot assistance with large language models. *Autonomous Robots*, 2023.
- [70] H. Xiong, R. Mendonca, K. Shaw, and D. Pathak. Adaptive mobile manipulation for articulated objects in the open world. *arXiv preprint arXiv:2401.14403*, 2024.

A Additional Real-world Task Visualization

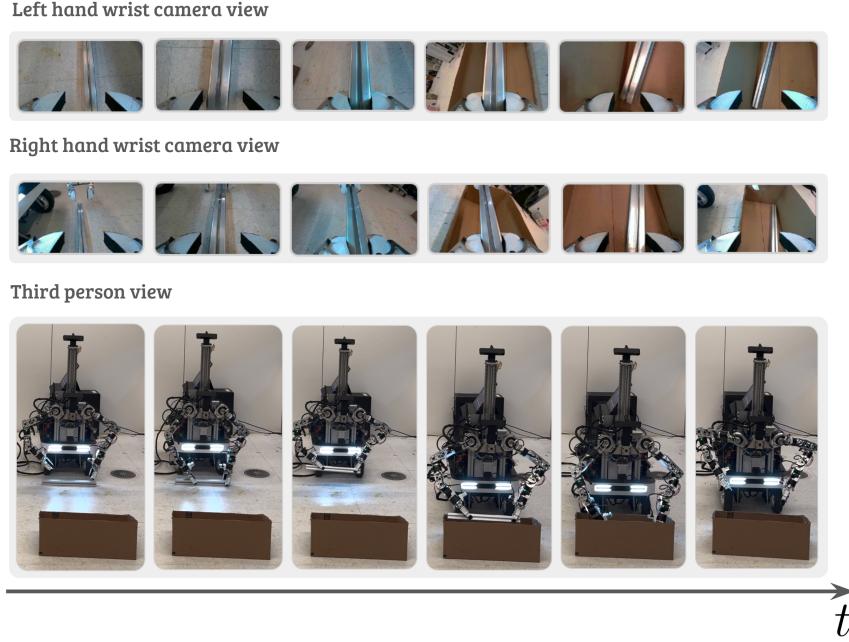


Figure 6: A step-by-step illustration of the industrial part transport task. The robot first needs to pick up the heavy industrial part from the floor using both hands. It then searches for and moves towards the tote. Finally, the robot must accurately drop the industrial part into the tote. The first two rows display the left and right wrist camera views along one autonomous trajectory. The bottom row shows a third-person view of the entire robot system during task execution.

B Simulation Task Visualizations

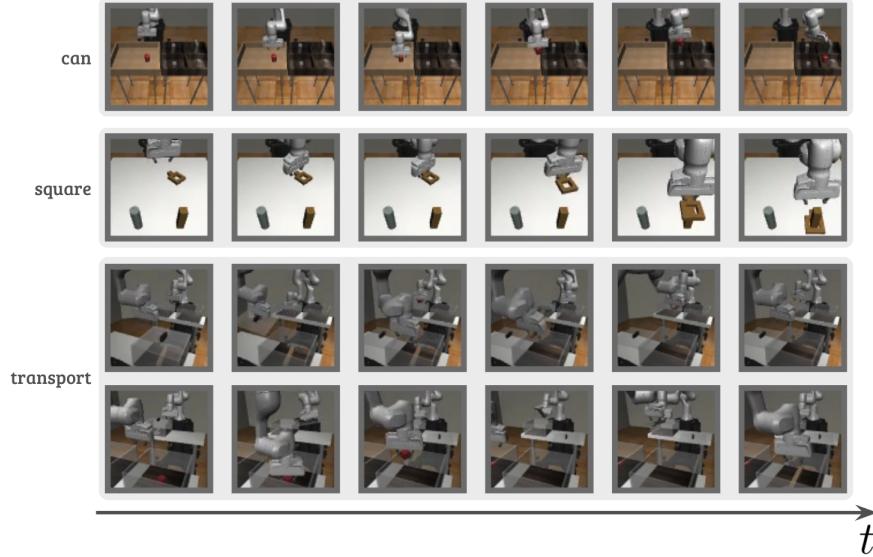


Figure 7: An illustration of the three manipulation tasks we use in Robomimic [14]: Pick up a can and place it in a bin; insert a square object into a pole; coordinate two arms to transport a tool.

C Teleoperation Capabilities

Our proposed RoboCopilot system not only enables efficient interactive teaching but also enhances teleoperation capabilities beyond the unilateral teleoperation devices [1, 3]. This improvement is achieved through active joint motor selection, which provides inertia compensation and bilateral feedback. Additionally, our fully compliant arm design allows for contact-rich bimanual tasks without compromising safety or risking damage to the robot hardware. In Figure 8 below, we demonstrate several daily kitchen tasks performed via teleoperation to showcase our system’s capabilities.

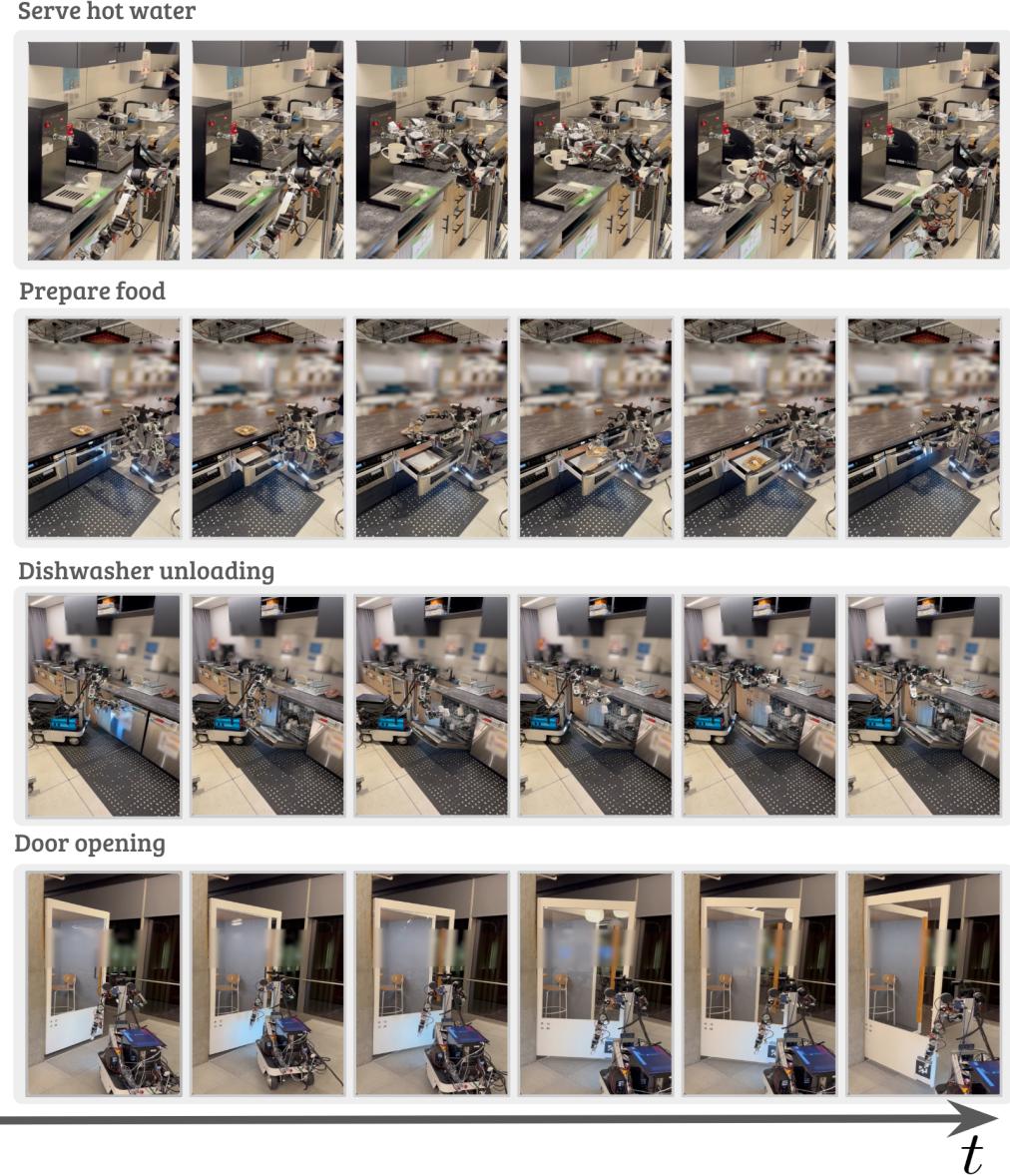


Figure 8: Teleoperated Execution of Various Contact-Rich and Whole-Body Control Tasks. The RoboCopilot system demonstrates its proficiency in performing a range of daily kitchen tasks under human teleoperation. These tasks include serving hot water, preparing food, unloading the dishwasher, and opening doors. The robot’s fully compliant design enables safe and effective handling of contact-rich interactions, showcasing its ability to perform a wide range of complicated tasks.

D Robot Hardware Details

The design space for such a complex system is large. Thus we utilize the following three key principles the development for our RoboCopilot system:

- **Safe:** The robot hardware must be safe for humans, the environment, and itself. Compliance, a mechanical property that describes the ability of the robot to respond to external forces, is critical to achieving this safety.
- **Easy to Use:** To facilitate efficient interactive learning, our system needs a capable teleoperation system that allows human operators to perform complex and contact-rich tasks. Additionally, this system must enable the human operator to take control of the robot seamlessly when necessary.
- **Accessible:** Our design aims to be cost-efficient and simple while still being able to perform a wide range of daily tasks. We believe accessibility is crucial for scaling up data collection and robot deployment in real-world scenarios.

Our robot, guided by the key principles above, is a custom low-cost mobile bimanual manipulator designed for everyday tasks. It consists of 7 DOF arms with parallel jaw grippers, an actuatable torso, and an omnidirectional base, totaling 20 degrees of freedom. We use two realsense D405 cameras [53] mounted on the wrist which are used to provide camera observations to the robot.

Compliance from Quasi-Direct Drive Actuators. Compliance allows the robot to feel when it is perturbed by an external force, which is crucial for the robot to safely work with humans and perform complicated tasks in environments where there is high uncertainty. Compliance can be achieved with the addition of force torque sensors by measuring and responding to external forces in a tight control loop [54, 55]. Compliance can also be achieved passively through a backdrivable transmission, where the system can be easily driven by external force without active control [56, 57, 58]. Taking cost and simplicity into consideration we achieve compliance with off-the-shelf mini-cheetah style [59] quasi direct drive (QDD) transmissions which achieve backdrivability through a low gear ratio planetary gearbox. In this setup, external forces applied at the gearbox output are transmitted to the motor and can be detected by measuring motor currents, where the motor can effectively act as a torque sensor.

Actuator Selection. We use off-the-shelf mini-cheetah style [59] brushless actuators with integrated planetary gearboxes Quasi Direct Drive (QDD) actuator as our robot joint modules. These QDDs typically have gear reduction ratios smaller than 10. With high torque brushless motors, these actuators provide sufficient torque for robotic applications while maintaining backdrivability. Although planetary gearboxes have backlash compared to gearboxes such as harmonic drives and cycloid drives, which may reduce precision, recent work utilizing end-to-end imitation learning with tight feedback loops suggests that low precision hardware can still be used for fine-grained tasks with feedback control [60, 61]. Furthermore, with the rise of legged robots applications, there are now widely available, affordable, and high-performing QDD options on the market. We are using 20Nm gear ratio of 9 actuators for the shoulder joints and 12Nm gear ratio of 40 actuators for the upper arm rotation and elbow joint and 3Nm gear ratio 10 actuators for the lower arm joints as well as the gripper joints, giving an approximate continuous payload of 1kg. The total cost of all 8 QDDs on each arm cost less than \$2000 for our design. For a more detailed cost breakdown, see Table 4.

Table 4: Robot cost breakdown per arm. Our 7 degrees of freedom arm consists of two 20Nm gear ratio 9:1 QDD(quasi-direct-drive) motors as the two shoulder actuators, two 12Nm gear ratio 40:1 QDD as the upper arm rotate and the elbow actuators and three 3Nm gear ratio 10:1 QDD for the lower arm rotate and the wrist yaw and pitch joints.

Part Name	QDD 20nm 9:1	QDD 12nm 40:1	QDD 3nm 10:1	Connectors	Total Price
Price Each	\$320	\$126	\$95	\$634	
Quantity	2	2	3	1 set	
Price	\$640	\$252	\$285	\$634	\$1811

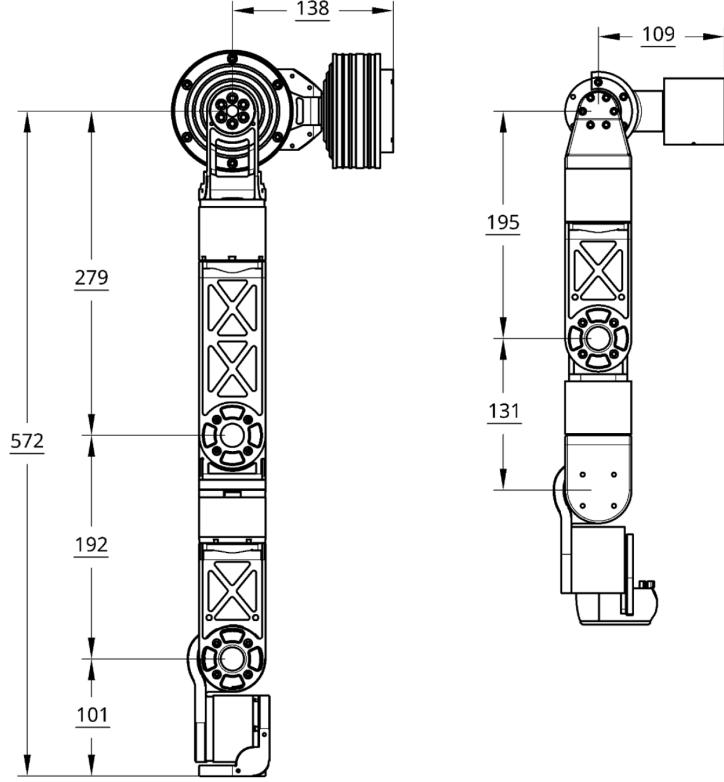


Figure 9: A to scale drawing of our robot arm (left) and teleoperation device (right), with dimensions shown in millimeters.

Arm Design. Figure 9 Shows the dimensions of the robot and the teleoperation device. The specs of our robot and teleoperation device are shown in Table 5. The bimanual system consists of replicating the robot arm and teleoperation device. We mount them at a 45 degree angle so that the first 2 joints can share the load of gravity. Following [3], we build a scaled kinematic replica of the arm as a teleoperation device and use the same motors we used for the robot arm for the teleoperation device. We set the scale to be 70% of the original size for easier human teleoperation. This however results in a design constraint, as the distance between joint 6 and joint 7 is 60mm, and with a motor diameter of 57mm, there leaves almost no room to scale down the teaching device. For this DOF we did not scale the kinematics, and found that despite not matching exactly, still resulted in a very capable and intuitive device.

Table 5: Various specifications about the robot and teleoperation device.

Robot	
Mass	4.6kg
Reach (Full Range)	572mm
Upper Arm Length	279mm
Lower Arm Length	252mm
Payload continuous	~ 1.5kg
Payload peak	~ 3.5kg
Degrees of Freedom	7

Teleoperation Device	
Mass	2.6kg
Upper Arm Length	195mm
Lower Arm Length	191mm
Degrees of Freedom	7

Gripper Design. Our high-speed gripper features an offset slider-crank linkage design. This design was chosen for simplicity and robustness over other low cost hands which have more complex mechanisms such as those presented by Guo et al. [62], Ma and Dollar [63], Shaw et al. [64]. The finger is designed to grasp on everyday sized objects. We implement layer jamming, leveraging 3D the printer slicer to generate Finray like geometry from our simple finger shape [65, 66, 67]. This design allows for efficient and reliable grasping of various objects. We leverage the same QDD actuators as our arm, which have a nominal RPM of 120, enabling us to achieve a continuous max gripping force of 25 N while being able to close and open under 1 second.

Mobility. A fully self contained mobile robot solution is important for ease of use and deployment in real world situations as well as the massive expansion of the robots workspace. Oftentimes, mobility in manipulation is achieved through using a mobile robot base, and mounting the rest of the robot system on top of it [68, 69, 56]. In line with the omni-directional base setup described in [70], we integrated our system with the off the shelf AgileX Ranger Mini 2 base, facilitating mobile bimanual manipulation. To expand the robot’s vertical operational space and enable the robot to pick up objects on the ground, the bimanual robot is mounted on a gantry, enabling vertical movement. The robot includes an onboard computer and router for Ethernet connectivity without Internet access. The entire mobile platform is powered by the Ranger Mini 2’s internal battery and an onboard 24v battery to make it truly wireless.