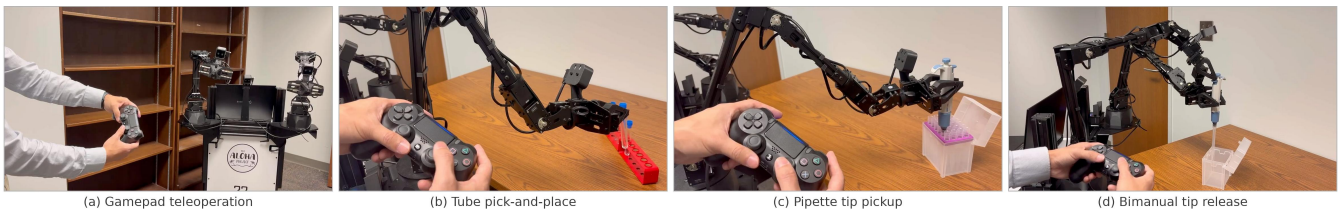
Leader-Free Mobile ALOHA:
Gamepad-IK Teleoperation and VLA Fine-Tuning for Laboratory Manipulation

Shengfeng Yang*
Purdue University
Project: https://shengfeng-yang.github.io/aloha-gamepad/



(a) Gamepad teleoperation (b) Tube pick-and-place (c) Pipette tip pickup (d) Bimanual tip release

Figure 1: Overview of the leader-arm-free teleoperation system for Mobile ALOHA. (a) Gamepad-IK teleoperation. (b) Tube pick-and-place. (c) Pipette tip pickup. (d) Bimanual pipette tip release. All tasks are performed by a single operator using the same gamepad interface.

Abstract

We present a leader-arm-free teleoperation system that uses a standard gamepad with inverse kinematics (IK) to control the full Mobile ALOHA platform—both bimanual arms and the mobile base. Unlike the standard leader-follower interface, which physically tethers the operator to the robot, our gamepad-IK approach decouples the operator, enabling them to walk close to the robot for visual precision and move freely around the workspace to find optimal viewing angles. We demonstrate this system on three laboratory manipulation tasks including precision pipette tip attachment, tube pick-and-place, and bimanual pipette tip release, showcasing the system's capability for both single-arm and coordinated bimanual control. Using 55 minutes of gamepad-collected demonstrations, we fine-tune the $\pi 0.5$ VLA and show that the resulting policy can correctly select the appropriate arm based on object position and successfully execute tube pickup, with robust visual tracking of object positions during execution. Code, videos, and dataset are available at: https://shengfeng-yang.github.io/aloha-gamepad/.

1 Introduction

Vision-Language-Action (VLA) models such as $\pi 0.5$ [1] can be fine-tuned with small demonstration datasets to learn new manipulation skills, making the quality and accessibility of demonstration collection increasingly important. The standard teleoperation interface for Mobile ALOHA [2] uses leader-follower puppeteering: the operator backdrives miniature leader arms while physically walking behind the mobile base. This provides intuitive kinesthetic control but tethers the operator to a fixed position relative to the robot.

For precision manipulation tasks—such as grasping a specific tube from a densely packed laboratory rack—this tethering is limiting. The operator cannot walk close to the end-effectors to observe fine contact details, cannot orbit the workspace to resolve occlusions, and cannot step back for a global view when needed. These limitations are not about control fidelity but about *perceptual access*: the operator's ability to see what matters during the demonstration.

We propose replacing leader-follower control with a wireless gamepad and an inverse kinematics (IK) solver that maps joystick inputs to Cartesian end-effector velocities (Figure 1). The primary advantage is not only dramatically lower cost—a standard PS4 gamepad costs ∼$30, replacing two WidowX 250 leader arms at $3,550 each (∼$7,100 total)—but also **operator mobility**: the operator is free to position themselves anywhere relative to the robot during demonstration. This enables:

- **Proximity.** Standing centimeters from the gripper-tube contact point during grasping, providing direct visual access to fine contact details that are invisible from the leader-follower operating position.
- **Walk-around viewpoints.** Physically orbiting the workspace to find the best viewing angle for each manipulation phase, mirroring how humans naturally adjust their head position when performing fine manual work.
- **Remote extensibility.** The same interface supports camera-mediated remote control for hazardous environments, as only wireless button/joystick state needs to be transmitted.

We demonstrate this system on a tube pickup task us-

*Corresponding author: yangshengfeng@gmail.com

ing the Mobile ALOHA platform with plastic laboratory equipment. We collect demonstrations, fine-tune $\pi 0.5$ using the LeRobot [3] training pipeline, and evaluate the resulting policy. All code, data, and evaluation scripts are released as open-source.

Contributions. (1) A complete gamepad + IK teleoperation system for Mobile ALOHA that controls bimanual arms and mobile base, with mode switching, Cartesian velocity control, and singularity handling, released as open-source at `https://github.com/shengfeng-yang/aloha-gamepad`. (2) An analysis of how operator mobility and proximity—enabled by leader-arm-free teleoperation—affect demonstration quality for precision manipulation tasks. (3) A demonstration that fine-tuning $\pi 0.5$ on 55 minutes of gamepad-collected data produces a functional policy capable of bimanual arm selection and tube pickup in a laboratory manipulation task.

2 Related Work

Leader-follower teleoperation for ALOHA. ALOHA [4] and Mobile ALOHA [2] use kinematically matched leader-follower arms for intuitive bimanual control. ALOHA 2 [5] improved ergonomics with low-friction grippers and a passive gravity compensation mechanism. Mobile ALOHA extends this to whole-body control by mounting the system on a wheeled base, with the operator physically tethered to the platform and backdriving the wheels for locomotion. While these systems provide high-quality demonstrations, the operator's position and viewpoint are fundamentally constrained by the leader-follower coupling.

Alternative teleoperation interfaces. AV-ALOHA [6] addresses the viewpoint problem by adding a 7-DoF active vision arm controlled via VR headset, but requires additional hardware and introduces VR-related motion sickness. TeleMoMa [7] provides a modular system supporting VR, vision-based, and combined interfaces for mobile manipulation. Gamepad teleoperation has been explored for single-arm systems: Prinz and Li [8] compared gamepad to leader-follower in a 36-participant user study, finding gamepads adequate for coarse manipulation; Pertsch et al. [9] used gamepads with an assistive policy for scalable multi-robot data collection, but their system controls only a single arm per gamepad—it does not support bimanual coordination or mobile base control. Interbotix provides built-in PS4 joystick control for the ALOHA platform [10], but this is limited to driving the mobile base alone; the arms must still be teleoperated via leader-follower puppeteering. In contrast, our system unifies control of both bimanual arms and the mobile base through a single gamepad, using modifier-based mode switching to provide access to all degrees of freedom without additional hardware. To our knowledge, this is the first gamepad-IK teleoperation system that provides complete whole-body control of a bimanual mobile manipulator from a single handheld controller, specifically designed for collecting high-quality VLA demonstration data.

VLA fine-tuning. $\pi 0$ [11] and $\pi 0.5$ [1] are generalist VLA models that can be fine-tuned on custom data via the openpi framework or the LeRobot [3] training pipeline. Physical Intelligence reported 1–20 hours of data sufficient for task adaptation [12]. Recent work has explored efficient fine-tuning with LoRA [13], mechanistic approaches [14] for few-shot adaptation, and efficient action tokenization [15] for improving VLA training. To our knowledge, this is the first work to fine-tune $\pi 0.5$ using demonstrations collected via gamepad-IK teleoperation.

3 System

Figure 2 provides an overview of the complete system pipeline, from gamepad teleoperation and data collection through VLA fine-tuning and autonomous policy execution.

3.1 Hardware

Our setup consists of: (a) a **Mobile ALOHA** platform with two ViperX 300 6-DoF follower arms on a mobile base, equipped with three Intel RealSense cameras (overhead, left wrist, right wrist); (b) a **PS4 DualShock 4** wireless gamepad connected via Bluetooth, read through the ROS2 joy node; and (c) **laboratory equipment**: an ONiLAB micropipette, a box of racked pipette tips, and plastic test tubes in a 6-slot red rack, all placed on a table within the robot's workspace. Figure 3 shows the Mobile ALOHA platform and the PS4 DualShock 4 gamepad used in our experiments.

3.2 Controller Mapping

We map the gamepad inputs to robot control across three functional groups: arm manipulation, base locomotion, and system utilities. Table 1 summarizes the complete mapping, and Figure 4 provides a visual diagram of the button and joystick assignments on the PS4 DualShock 4 controller.

The mapping is designed around three principles. First, **modal arm selection**: L1 and R1 select the active arm, with L1+R1 enabling simultaneous bimanual control, allowing the operator to focus joystick commands on one arm at a time. Second, **modifier-based mode switching**: holding L2 redirects the right stick to base control, and holding R2 redirects the left stick to orientation control. This design balances the workload between the operator's left and right hands—position control (left stick) and height/roll control (right stick) are distributed across both hands by default, while the modifier buttons allow either hand to temporarily take on an additional role (base or orientation) without requiring the operator to release the other stick. This keeps the mapping

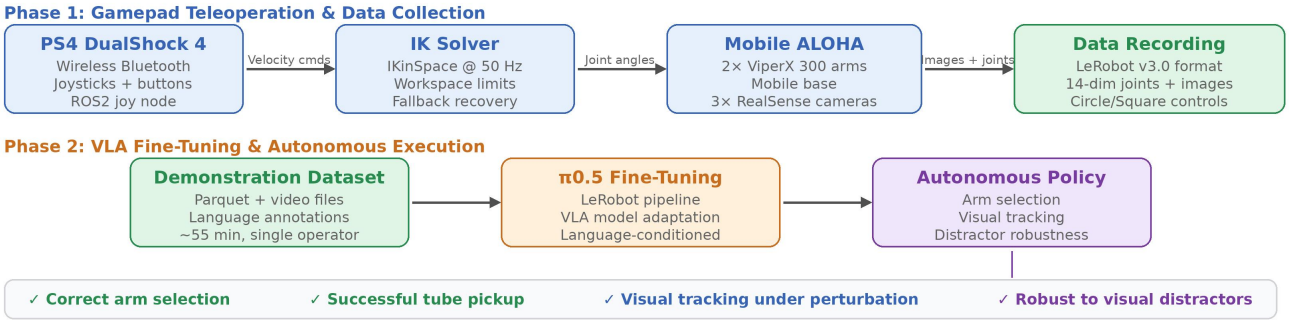


Figure 2: System pipeline overview. Phase 1: the operator uses a PS4 DualShock 4 gamepad to control the Mobile ALOHA via inverse kinematics and records demonstrations in LeRobot v3.0 format. Phase 2: the demonstration dataset is used to fine-tune $\pi 0.5$ for autonomous execution.

Table 1: Complete gamepad control mapping. Arm selection via L1/R1 determines which arm receives manipulation commands.

Control	Action
Left Stick	EE forward/back, left/right
Right Stick	EE up/down, roll
R2 + Left Stick	Pitch / Yaw orientation
L2 + Right Stick	Base forward/back + rotation
Triangle / X	Open / Close gripper
L1 / R1	Select left / right arm
D-Pad Up/Down	Speed adjustment
Circle	Start / Stop recording
Square	Discard current recording
Options / Share	Home pose (one / both arms)
PS Button	Exit program

compact without sacrificing degrees of freedom. Third, **integrated recording**: Circle toggles episode recording and Square discards a bad take, enabling seamless data collection without leaving the teleop interface.

3.3 Inverse Kinematics

At each control step (50 Hz), we read the joystick state and compute a desired Cartesian velocity twist. Position deltas (dx, dy, dz) are derived from stick deflections scaled by a configurable speed gain (adjustable at runtime via D-Pad Up/Down, range 10%–100% in 10% increments) and the linear scale factor of 0.1 m/s. Orientation deltas ($d\phi_{\text{roll}}$, $d\phi_{\text{pitch}}$, $d\phi_{\text{yaw}}$) use an angular scale of 0.5 rad/s. These deltas are integrated into a target end-effector pose, and we solve for joint positions using the Modern Robotics `IKinSpace` solver with the ViperX 300s screw axes and home configuration matrix. The solver uses an orientation tolerance of 0.01 rad and a position tolerance of 1 mm. The current joint configuration seeds each IK call for continuity.

3.4 Workspace Limits

Operating a 6-DoF arm via Cartesian commands creates a risk of commanding the end-effector outside the reachable workspace, leading to IK failures, joint limit violations, or abrupt motions near singularities. To address this, we enforce a three-layer workspace protection system.

First, we apply rectangular Cartesian bounds that constrain the end-effector position to a box defined by the ViperX 300s kinematic specifications (750 mm reach, 1500 mm total span): $x \in [0.10, 0.65]$ m (forward/backward), $y \in [-0.50, 0.50]$ m (left/right), and z up to 0.45 m. Rather than using a fixed z minimum, we implement a distance-dependent lower bound: when the arm is extended ($x = 0.65$ m) the end-effector can reach $z = -0.05$ m (below the table surface), but when retracted ($x = 0.10$ m) the minimum z is raised to 0.20 m. This linear interpolation reflects the physical reality that a retracted arm cannot safely reach downward without self-collision.

Second, every IK solution is validated against per-joint limits derived from the ViperX 300s URDF, with safety buffers of 20° for high-range joints (waist, forearm roll, wrist rotate) and 5° for more constrained joints (shoulder, elbow, wrist angle). Solutions that violate any joint limit are rejected even if the IK solver converges.

Third, all orientation angles are wrapped to $[-\pi, \pi]$ after each update to prevent numerical drift from accumulating unbounded angles.

These protections are critical for gamepad teleoperation because, unlike leader-follower control where the operator's own joint limits naturally prevent unreachable commands, a joystick can trivially command poses outside the workspace. The workspace limits allow the operator to push against boundaries safely—the end-effector simply stops at the limit while the system prints a real-time warning message on the controlling laptop screen identifying which arm hit the limit (e.g., "RIGHT arm hit workspace limit"). To avoid flooding the console, the warning is printed once on the first hit and then periodically every 50 control cycles during sustained contact

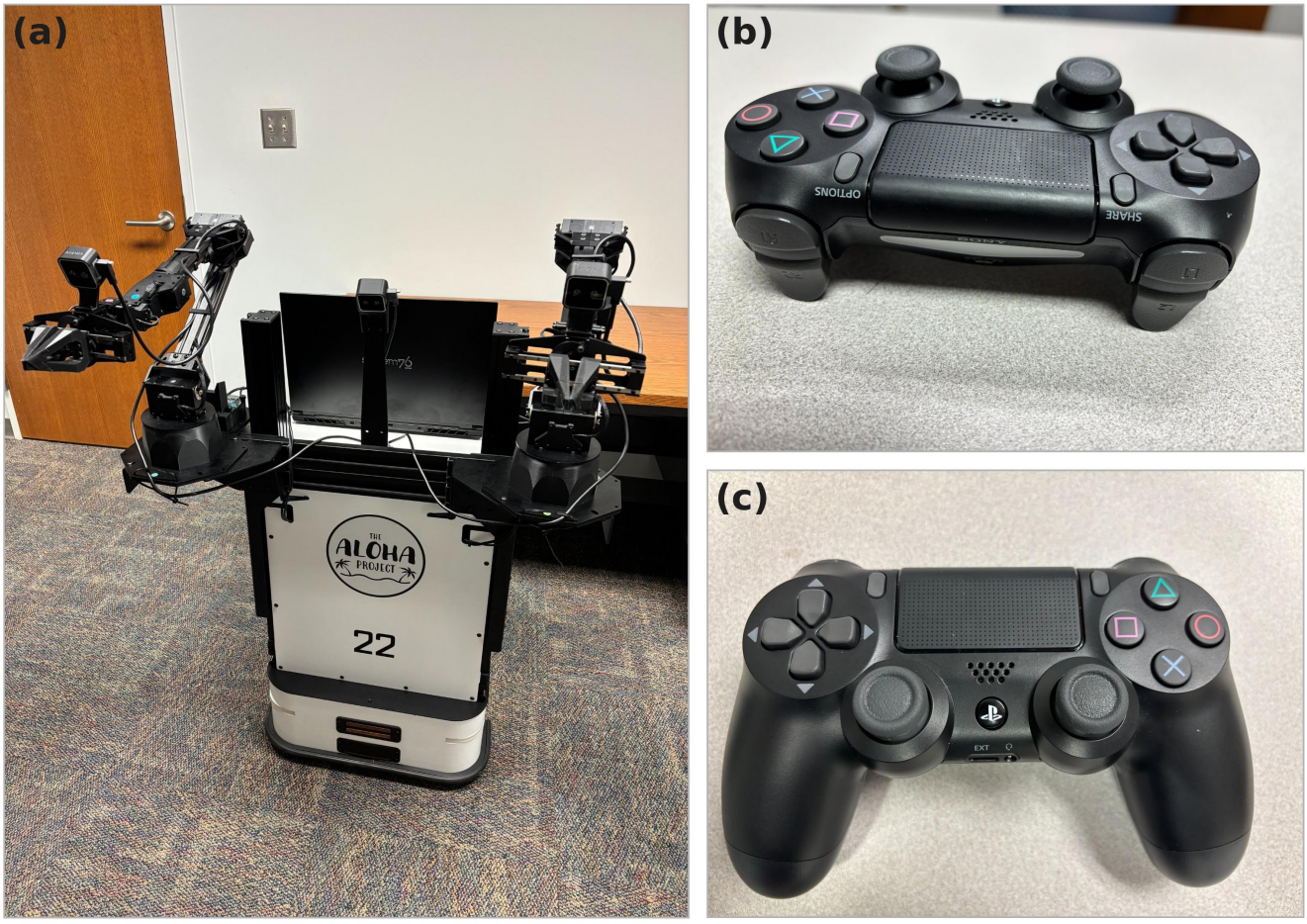


Figure 3: Hardware setup. (a) The Mobile ALOHA platform with bimanual ViperX 300 arms, onboard System76 laptop, and mobile base. (b–c) The PS4 DualShock 4 wireless gamepad used for leader-free teleoperation.

with the boundary. Because all workspace parameters are defined as constants at the top of the teleoperation script, they can be easily adjusted by the user to accommodate different table heights, task-specific reach requirements, or alternative arm configurations without modifying any control logic.

3.5 IK Fallback and Recovery

Even with workspace limits, IK can fail when the target pose is kinematically unreachable due to orientation constraints or proximity to singularities. We implement a progressive fallback strategy to handle these failures gracefully.

When IK fails, the system holds the arm at its last successfully commanded joint configuration, preventing any uncontrolled motion. A consecutive failure counter tracks how many successive IK calls have failed. After 5 consecutive failures (`IK_MAX_CONSECUTIVE_FAILS`), the system activates a recovery mechanism: it shifts the target end-effector pose 30% back toward the last known good pose (`IK_FALLBACK_FACTOR = 0.3`). This gradual pull-back typically moves the target into a reachable region within a few control cycles, at which point IK succeeds again and normal control resumes.

The fallback operates independently per arm, so a failure on one arm does not affect the other. Throughout this process, the system provides clear real-time feedback on the controlling laptop screen: a warning is printed on the first IK failure identifying the affected arm and the reason (e.g., "IK failed for right arm (target out of reach)" or "IK solution exceeds shoulder joint limit"), a notification when the fallback mechanism activates ("IK fallback: right arm moving back toward reachable zone"), and periodic status updates during sustained failures. This on-screen feedback loop is essential because, unlike leader-follower control where the operator feels physical resistance at the limits, gamepad control provides no haptic feedback—the laptop screen serves as the primary channel for communicating the arm's kinematic state to the operator.

This mechanism is essential for practical gamepad teleoperation. When an operator pushes the arm into an awkward configuration—for example, attempting to reach be-

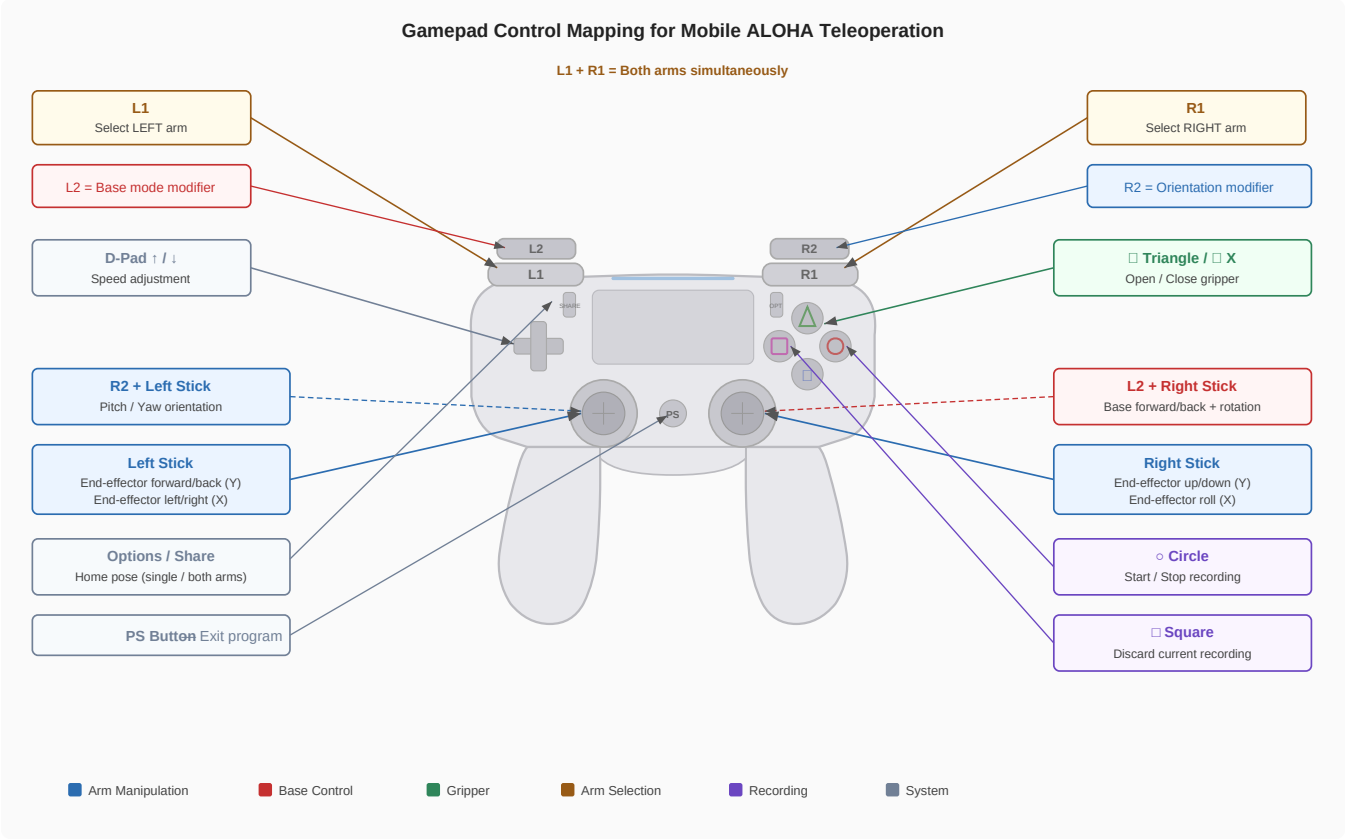


Figure 4: Gamepad diagram with labeled button/joystick mappings.

hind an obstacle—the system degrades gracefully rather than freezing or jumping. The operator simply reverses their joystick input and the arm smoothly re-enters the reachable workspace.

3.6 Base Acceleration Limits and Safety

Mobile base control requires particular care because sudden velocity changes can destabilize the platform, disturb the arms' manipulation, or create unsafe conditions for nearby operators. We implement three safety measures for base control.

First, the base is gated behind a deadman switch: the L2 button must be held to enable base movement, and releasing L2 immediately commands zero velocity. This prevents accidental base motion during arm teleoperation, which uses the same right stick.

Second, we apply acceleration limiting to smooth velocity transitions. The maximum linear acceleration is capped at $0.15\,\mathrm{m/s^2}$ and angular acceleration at $0.5\,\mathrm{rad/s^2}$. At each 50 Hz control cycle, the velocity change is clamped to these limits, producing smooth ramps rather than step changes. This is implemented as:

$$v(t) = v(t{-}1) + \mathrm{clamp}(v_t - v(t{-}1), \pm a_{\max}\Delta t) \qquad (1)$$

where v_t is the target velocity, $a_{\max}$ is the acceleration limit, and $\Delta t = 0.02\,\mathrm{s}$. The maximum linear velocity is $0.2\,\mathrm{m/s}$ and the maximum angular velocity is $0.4\,\mathrm{rad/s}$, both further scaled by the operator's speed setting.

Third, the velocity command is sent directly to the base at the full 50 Hz control rate for responsive stopping behavior, ensuring that a release of L2 results in a near-immediate halt.

These measures are particularly important for the leader-free teleoperation paradigm: because the operator may be standing next to the robot rather than behind it, smooth and predictable base motion is essential for safety. The acceleration limits also produce smoother trajectories in the recorded demonstrations, reducing discontinuities that could harm downstream policy learning.

3.7 Data Recording

Demonstrations are recorded in **LeRobot v3.0 format** for direct compatibility with the LeRobot [3] training pipeline and $\pi 0.5$ fine-tuning. The recording pipeline is implemented in a dedicated `lerobot_writer.py` module that handles serialization to Apache Parquet with video encoding via imageio-ffmpeg. Each timestep records: RGB images from three cameras (overhead, left wrist, right wrist), joint positions (14-dim: 6 joints + 1 gripper $\times$ 2 arms), base velocity (2-dim: linear, angular), target action commands, and a language instruction string (e.g.,

"pick up the tube from the rack").

Recording is toggled via the Circle button on the gamepad. If a demonstration is unsatisfactory, the operator presses Square to discard the current episode without saving. This integrated workflow—teleoperate, record, review, discard or keep—enables efficient data collection without switching between separate tools.

3.8 Software Architecture

The system is implemented as a set of Python scripts built on ROS2 Humble and the Interbotix ROS packages. The main teleoperation loop (`gamepad_teleop.py`) handles gamepad input, IK solving, arm command publishing, and recording orchestration. A separate launch file (`aloha_bringup.launch.py`) brings up the ALOHA hardware nodes including follower arms, cameras, and mobile base drivers. Additional utilities include `sleep_arms.py` for safely parking the arms after data collection, and `view_dataset.py` for visualizing collected episodes with per-camera playback and dataset summary statistics. The complete codebase is available at https://github.com/shengfeng-yang/aloha-gamepad.

4 Experiments

We evaluate our gamepad-IK teleoperation system through three demonstration tasks that showcase the system's capability for precision laboratory manipulation, including both single-arm and bimanual coordination, followed by a large-scale data collection experiment that validates the downstream utility of gamepad-collected demonstrations for VLA fine-tuning. Figure 5 shows the three demonstration tasks with the operator controlling the robot via the PS4 gamepad.

4.1 Demonstration Task 1: Pipette Tip Pickup

In the first task (Figure 5a), the robot holds a pipettor (ONiLAB P5000 micropipette) in its gripper and must press the pipettor tip into a box of pipette tips to attach a new tip. This task requires precise vertical alignment and controlled downward force—the pipettor must be positioned directly above an empty slot in the tip box and pressed down firmly enough to seat the tip but not so hard as to damage the box or the pipettor. The operator uses the gamepad to control the arm holding the pipettor, leveraging the leader-free mobility advantage to stand directly above the tip box and visually verify alignment before initiating the press.

4.2 Demonstration Task 2: Tube Pick-and-Place

In the second task (Figure 5b), the robot must use its arm and gripper to pick up a test tube with a blue cap from a 6-slot red plastic rack, lift it, and place it back into the rack. This task tests the system's precision for grasping small cylindrical objects from a constrained holder and the control fidelity required for re-insertion. The operator benefits from the walk-around viewpoint capability, moving to different angles to verify the gripper alignment with the tube during both the grasp and the placement phases.

4.3 Demonstration Task 3: Bimanual Pipette Tip Release

The third task (Figure 5c) demonstrates the system's capability for coordinated bimanual manipulation using a single gamepad. One arm holds the pipettor (with a tip already attached), while the other arm reaches over and presses the tip release button on the pipettor to eject the used tip. This task requires precise coordination between the two arms: the holding arm must keep the pipettor stable while the other arm locates and presses the small release button with sufficient force. The operator uses the L1/R1 arm selection to switch control between the two arms, first positioning the holding arm, then switching to the pressing arm for the button press. This task highlights a key advantage of our unified gamepad-IK interface: both arms are controlled through the same controller with seamless switching, enabling bimanual coordination that would otherwise require two separate operator inputs or a more complex teleoperation setup.

4.4 Data Collection for π0.5 Fine-Tuning

To evaluate the downstream utility of gamepad-collected demonstrations for VLA fine-tuning, we conducted a large-scale data collection session focused on the tube pick-and-place task. The tube and rack were placed at varying positions within the workspace, close to either the left arm or the right arm, requiring the policy to learn both which arm to use and how to execute the grasp. We collected approximately 55 minutes of teleoperated demonstrations using the gamepad-IK interface, recording in LeRobot v3.0 format.

During data collection, the operator leveraged the mobility advantages of leader-free teleoperation: standing beside the robot to observe the arm trajectory during approach, moving within 10–15 cm of the gripper during the grasp phase to verify alignment, and stepping to the side to check grasp quality before placing the tube back. Unsatisfactory demonstrations were immediately discarded using the Square button on the gamepad. The integrated recording workflow—teleoperate, record, review, discard or keep—enabled efficient data collection without switching between separate tools.

(a) Pipette tip pickup (b) Tube pick-and-place (c) Bimanual tip release

Figure 5: Three demonstration tasks performed with gamepad-IK teleoperation. (a) Pipette tip pickup: the arm holds a pipettor and presses it into a tip box to attach a tip. (b) Tube pick-and-place: the arm grasps a test tube from a rack. (c) Bimanual pipette tip release: one arm holds the pipettor while the other presses the release button to eject the tip.

4.5 Fine-Tuning $\pi 0.5$

We fine-tune $\pi 0.5$ using the LeRobot training pipeline on the 55 minutes of collected demonstrations. The data is already in LeRobot v3.0 format, requiring no format conversion. The language prompt used for conditioning is "pick up the tube from the rack." Because the tube and rack were placed near either arm during data collection, the model must learn to select the appropriate arm based on the visual observation and then execute the grasp.

5 Results

We evaluate the fine-tuned $\pi 0.5$ policy on the tube pickup task to demonstrate that the gamepad-collected demonstrations are sufficient for training a functional manipulation policy. The tube and rack are placed at varying positions near either the left or right arm, requiring the policy to both select the appropriate arm and execute the grasp.

The fine-tuned policy demonstrates several key capabilities that validate the quality of gamepad-collected demonstrations:

Arm selection. The policy reliably selects the arm closest to the tube based on visual observation. When the tube is placed near the left arm, the policy activates the left arm; when placed near the right arm, it activates the right. This spatial reasoning capability emerges naturally from the bimanual demonstrations collected with our system, where the operator controlled both arms through the same interface, producing demonstrations that include both left-arm and right-arm task executions.

Tube pickup. Once the correct arm is selected, the policy executes a grasp trajectory that successfully picks up the tube from the rack in the majority of trials. The demonstrations collected via the gamepad-IK interface provide sufficient quality for the model to learn precise grasping of small cylindrical objects from constrained holders.

Visual tracking. We observe that the fine-tuned policy exhibits robust visual tracking (Figure 6): when the tube position is moved during testing—including both relocating the rack and changing which slot the tube occupies—the policy dynamically adjusts its arm trajectory to follow the new tube location rather than reaching toward the original position. As shown in Figure 6, even after a human physically moves the rack and reinserts the tube into a different slot mid-execution (step 3), the policy retargets to the new position (step 4) and successfully completes the pickup (step 5). This indicates that the model has learned a reactive, vision-based grasping strategy rather than memorizing fixed trajectories from the demonstration data—a property that is essential for deployment in real laboratory environments where object positions may vary.

These results are notable because the entire 55-minute dataset was collected by a single operator using a standard wireless gamepad, without any leader-follower hardware. The focus of this work is not on optimizing the fine-tuned model's performance, but on demonstrating that the gamepad-IK teleoperation system produces demonstrations of sufficient quality to train functional VLA policies for precision laboratory manipulation tasks.

Robustness to visual distractors. We further evaluate the fine-tuned policy in the presence of background clutter not seen during training. As shown in Figure 7, we place distractor objects (spray bottles) near the tube rack during policy execution. Despite these novel visual distractors, the policy correctly identifies and approaches the target tube, ignoring the irrelevant objects, and successfully completes the pickup. This demonstrates that the policy has learned to focus on the task-relevant features (tube and rack) rather than overfitting to the backgrounds present in the training demonstrations.

Video demonstrations of all three teleoperation tasks (pipette tip pickup, tube pick-and-place, and bimanual pipette tip release) as well as the autonomous $\pi 0.5$ policy rollouts including the position tracking behavior are available on the project page at https://shengfeng-yang.github.io/aloha-gamepad/.

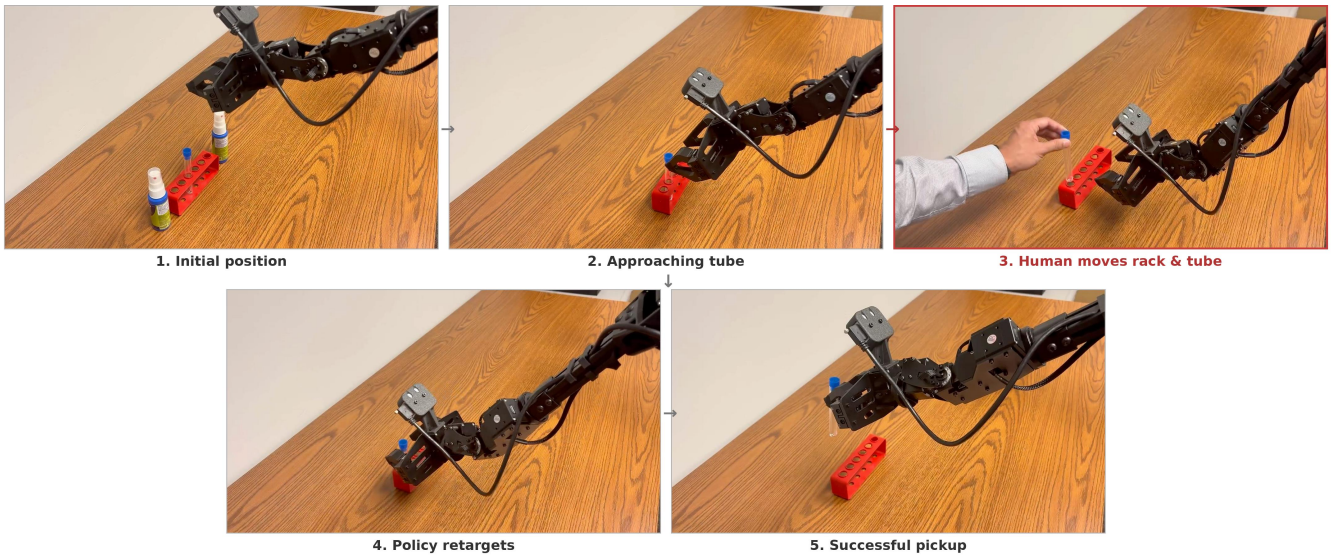


Figure 6: Visual tracking demonstration of the fine-tuned $\pi 0.5$ policy. (1) The arm begins in its initial position with the tube in the rack. (2) The policy approaches the tube for grasping. (3) A human intervenes and moves the rack to a new position while also changing the tube's slot (highlighted in red). (4) The policy retargets to the new tube location without restarting. (5) The arm successfully picks up the tube from its new position.

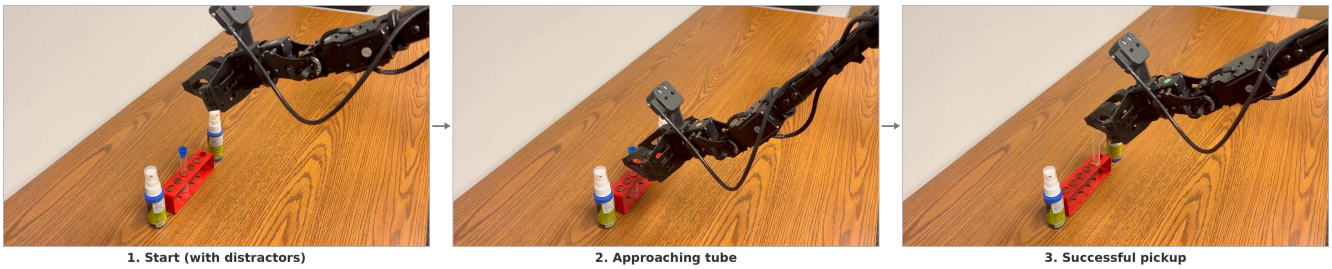


Figure 7: Robustness to visual distractors. The fine-tuned $\pi 0.5$ policy successfully picks up the tube despite the presence of distractor objects (spray bottles). (1) Start with distractors placed near the rack. (2) The policy approaches the correct tube, ignoring distractors. (3) Successful pickup.

6 Discussion

When does leader-free control help? The mobility advantage is most pronounced for tasks requiring visual precision at the contact point—exactly the scenario in laboratory manipulation. For large-motion tasks where the operator's natural viewpoint from behind the robot is adequate (e.g., opening cabinets, swiping), leader-follower may remain more efficient due to its more direct kinesthetic mapping.

Control bandwidth tradeoff. Gamepad joysticks provide lower control bandwidth than direct kinesthetic puppeteering. However, the ability to see the contact point up close partially compensates, as the operator makes fewer corrective attempts. The runtime speed adjustment via D-Pad further helps: operators use high speed for coarse motion and low speed for fine alignment.

Extensibility. The gamepad-IK interface naturally supports: (1) remote teleoperation via camera streaming for hazardous lab environments, (2) shared autonomy where the operator provides high-level guidance while the policy handles fine control, and (3) multi-robot data collection where one operator switches between robots with a single controller.

Integrated workflow. A practical advantage of our system is the fully integrated data collection workflow. The operator can teleoperate, start/stop recording, discard bad episodes, and visualize collected data without leaving the gamepad interface or switching tools. This reduces friction in the data collection loop and makes it easy to rapidly iterate on demonstration quality.

7 Limitations

Several limitations suggest directions for future work. First, gamepad joysticks are inherently lower-bandwidth than direct kinesthetic control via leader arms; the modifier-based orientation control (R2 + Left Stick for pitch/yaw) adds an extra step compared to the direct

6-DoF mapping of leader-follower. For tasks demanding very fast bimanual coordination (e.g., knot tying), leader-follower puppeteering may be preferred. Second, while gamepad familiarity is widespread, the specific control mapping requires a learning period for basic competency. Third, we evaluate on a single precision manipulation task; generalization to other laboratory tasks (pipetting, sample transfer) and non-laboratory domains remains to be demonstrated. Finally, while our interface supports remote teleoperation in principle, we have not yet evaluated the effect of camera-mediated operation and network latency on demonstration quality.

8 Conclusion

We presented a leader-free teleoperation system for Mobile ALOHA that uses a standard gamepad with inverse kinematics to control bimanual arms and a mobile base. By decoupling the operator from the robot, our system enables operator mobility—the ability to walk close to the robot for visual precision and orbit the workspace for optimal viewpoints. We demonstrated that 55 minutes of demonstrations collected with this interface are sufficient to fine-tune $\pi 0.5$ for a bimanual laboratory tube pickup task, producing a policy capable of correct arm selection, successful grasping, and robust visual tracking of object positions. The complete system—including teleoperation, data recording in LeRobot format, policy evaluation, and dataset visualization tools—is released at https://github.com/shengfeng-yang/aloha-gamepad.

References

[1] Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025.

[2] Zipeng Fu, Tony Z. Zhao, and Chelsea Finn. Mobile ALOHA: Learning bimanual mobile manipulation with low-cost whole-body teleoperation. In *Proceedings of The 8th Conference on Robot Learning (CoRL)*, volume 270 of *Proceedings of Machine Learning Research*, pages 4066–4083, 2024.

[3] Remi Cadene, Simon Alibert, Alexander Soare, Quentin Gallouedec, Adil Zouitine, Steven Palma, Pepijn Kooijmans, Michel Aractingi, Mustafa Shukor, Dana Aubakirova, Martino Russi, Francesco Capuano, Caroline Pascal, Jade Choghari, Jess Moss, and Thomas Wolf. LeRobot: State-of-the-art machine learning for real-world robotics in pytorch. https://github.com/huggingface/lerobot, 2024.

[4] Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Robotics: Science and Systems (RSS)*, 2023.

[5] ALOHA 2 Team, Jorge Aldaco, Travis Armstrong, Robert Baruch, Jeff Bingham, Sanky Chan, Kenneth Draper, Debidatta Dwibedi, Chelsea Finn, Pete Florence, Spencer Goodrich, Wayne Gramlich, Torr Hage, Alexander Herzog, Jonathan Hoech, Thinh Nguyen, Ian Storz, Baruch Tabanpour, Leila Takayama, Jonathan Tompson, Ayzaan Wahid, Ted Wahrburg, Sichun Xu, Sergey Yaroshenko, Kevin Zakka, and Tony Z. Zhao. ALOHA 2: An enhanced low-cost hardware for bimanual teleoperation. *arXiv preprint arXiv:2405.02292*, 2024.

[6] Ian Chuang, Andrew Lee, Dechen Gao, M-Mahdi Naddaf-Sh, and Iman Soltani. Active vision might be all you need: Exploring active vision in bimanual robotic manipulation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7952–7959, 2025.

[7] Shivin Dass, Wensi Ai, Yuqian Jiang, Samik Singh, Jiaheng Hu, Ruohan Zhang, Peter Stone, Ben Abbatematteo, and Roberto Martín-Martín. TeleMoMa: A modular and versatile teleoperation system for mobile manipulation. *arXiv preprint arXiv:2403.07869*, 2024.

[8] Theresa Prinz and Jinyang Li. Design and evaluation of a gamepad-based control scheme for the teleoperation of a stationary robot. In *Human-Computer Interaction – HCI International 2025 (HCII)*, volume 15772 of *Lecture Notes in Computer Science*, pages 138–154. Springer, 2025.

[9] Shivin Dass, Karl Pertsch, Hejia Zhang, Youngwoon Lee, Joseph J. Lim, and Stefanos Nikolaidis. PATO: Policy assisted teleoperation for scalable robot data collection. In *Robotics: Science and Systems (RSS)*, 2023. arXiv preprint arXiv:2212.04708.

[10] Trossen Robotics. ALOHA documentation: Mobile operation. https://docs.trossenrobotics.com/aloha_docs/, 2024.

[11] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey

Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control. In *Robotics: Science and Systems (RSS)*, 2025. arXiv preprint arXiv:2410.24164, 2024.

[12] Physical Intelligence. Open sourcing π_0. https://github.com/Physical-Intelligence/openpi, 2025.

[13] Abdullah Yahya Abdullah Omaisan and Ibrahim Sheikh Mohamed. Towards accessible physical AI: LoRA-based fine-tuning of VLA models for real-world robot control. *arXiv preprint arXiv:2512.11921*, 2025.

[14] Chancharik Mitra, Yusen Luo, Raj Saravanan, Dantong Niu, Anirudh Pai, Jesse Thomason, Trevor Darrell, Abrar Anwar, Deva Ramanan, and Roei Herzig. Mechanistic finetuning of vision-language-action models via few-shot demonstrations. *arXiv preprint arXiv:2511.22697*, 2025.

[15] Karl Pertsch, Kyle Stachowicz, Brian Ichter, Danny Driess, Suraj Nair, Quan Vuong, Oier Mees, Chelsea Finn, and Sergey Levine. FAST: Efficient action tokenization for vision-language-action models. *arXiv preprint arXiv:2501.09747*, 2025.