Teaching an SO-101 arm to play chess

Hakim El Ayoubi
University College London

Abstract

The SO-101 is a low-cost, open-source arm sold as a leader–follower pair for teleoperation. This report describes an attempt to make it play black against a human with a single wrist-mounted camera as its only view of the board. Stockfish selects each move, and a policy fine-tuned from SmolVLA, a 450-million-parameter vision-language-action model, grasps and places the piece. I recorded 202 teleoperated demonstrations, 54 minutes in total, and evaluated each configuration on a fixed set of 20 single moves. End-to-end fine-tuning completed 3 of the 20 moves. Drawing the target into the camera image completed 5, and splitting each move into a scripted approach, carry and return around two learned skills, grasp and place, completed 8. With 20 trials per configuration none of these differences is statistically significant.

Most of the errors that could be measured directly were geometric. The hand-eye calibration started at 146.6 px of reprojection error, and in the arm's kinematic frame the board measured about 9 % narrower across the files than its true width. A grasp-only drill of 237 demonstrations raised the per-attempt grasp rate from 41 % to 59 %, but complete moves did not improve, at 7 of 22, because the place stage lost 5 of the 12 pieces it received.

I then matched the follower's deployed gain to the one used for most drill recordings, recorded 120 grasp and 100 placement corrections at states the policy reached, and fine-tuned one checkpoint for both skills. With that checkpoint frozen, the final evaluation gave 190 of 200 first-attempt grasps (95.0 %) and 195 of 200 first-attempt placements of a held piece. Of 200 complete transfers, 184 succeeded without a retry and 194 within the recovery budget. In eight games from the standard starting position, human white against robot black, six reached a legal outcome without manual correction and two finished after one correction each. The games are short and the sample is small.

1 Introduction

The task is for an SO-101 arm to play black in a normal game against a human, with its wrist camera as the only view of the board. Stockfish selects each move; the problem studied here is its execution, from the instruction "move the black knight from g8 to f6" to the knight standing on f6 with no other piece disturbed. On the board used here the squares are 28 mm apart, so a grasp 20 mm off target is 8 mm from the neighbouring piece. Execution is also required to tolerate a shift of 5 to 10 % in the board or piece positions, 11 to 22 mm on this 224 mm board. The manipulation is learned by fine-tuning SmolVLA on teleoperated demonstrations rather than programmed.

Leader–follower teleoperation of low-cost arms was introduced to robot learning with ALOHA and ACT [3]; the SO-100 and SO-101 arms and the LeRobot library [2] reduced the cost further. Vision-language-action models such as RT-2 [7], OpenVLA [8], π_0 [9] and SmolVLA [1] map an image and an instruction to actions. Like Diffusion Policy [10] and ACT, they predict chunks of future

Project report, November 2026. Code and data: private for now.

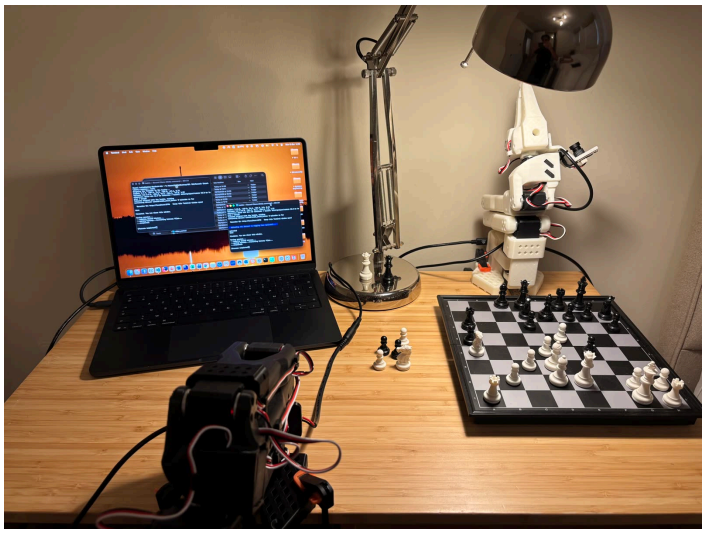


Figure 1. The rig. The SO-101 follower arm stands beside the board, with the leader arm used for teleoperation in the foreground and the laptop that runs the player behind.

actions, and real-time chunking [11] addresses the pauses between chunks described in Section 5.4. DAgger [12] and HG-DAgger [13] collect corrections in the states where a policy fails. Earlier chess-playing manipulators, Gambit [14] among them, used engineered perception and motion planning. The camera calibration uses the equidistant fisheye model [15] and OpenCV's hand-eye solvers [6, 16].

Section 2 describes the hardware and the MuJoCo model of the arm, Section 3 the camera calibration, Section 4 the demonstrations and Section 5 end-to-end fine-tuning. Section 6 splits each move into scripted and learned stages, Section 7 calibrates the board in the arm's frame, Section 8 adds a grasp-only drill and Section 9 collects the development evaluations. Section 10 matches the deployed system to the demonstrations and adds corrections, Section 11 describes the game player, Section 12 the final evaluation and Section 13 discusses the results. All numbers come from the project journal, which records each step with its measurements. The charts are generated from the project's logs, and the tables in Section 12 reproduce the journal's.

2 Hardware and simulation

2.1 The arm

The SO-101 consists of a follower arm and a leader arm with the same kinematics; the leader is moved by hand. In teleoperation the follower tracks the leader joint by joint, and demonstrations are recorded this way (Figure 2). The state of each arm is six joint positions, the gripper included. The software is LeRobot 0.6.2, installed from source, on a MacBook Air M4, which also runs inference; training runs on rented GPUs. The wrist camera has a fisheye lens with a horizontal field of view of 88.8° and a vertical one of 49.9°. It is mounted beside the gripper, and the jaw covers the right 40 % of the image.

Three set-up issues affected later work. The leader runs on a 5 V supply and the follower on 12 V, and a leader motor that appeared to be missing was powered from the wrong supply. Conda installed ffmpeg 9, which broke video decoding in torchcodec, so ffmpeg is pinned at 7.1.1. LeRobot places each joint's zero at the middle of the range swept during calibration, so an incomplete sweep shifts the zero; the resulting offsets reappear in the hand-eye calibration (Section 3.2).

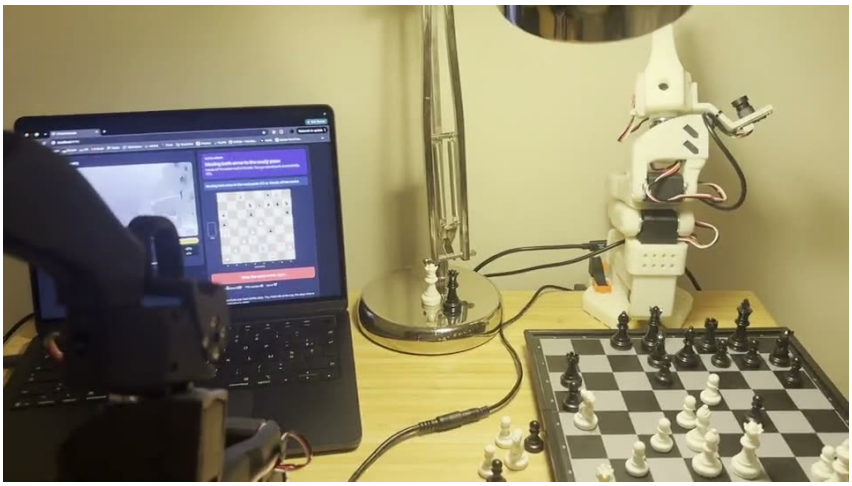


Figure 2. Teleoperation. The dark leader arm in the foreground is moved by hand and the white follower copies it; the recorder on the laptop shows the follower's wrist camera.

2.2 The MuJoCo model

The MuJoCo [4] model of the rig contains an SO-101 MJCF, the wrist camera, a procedurally generated board with real piece dimensions and a drop zone for captured pieces. The board is placed 10 cm from the base origin with rank 8 nearest the arm, and the drop zone lies at least 4.6 cm to the robot's right of the board. MuJoCo was chosen over Isaac Sim because it runs natively on the Mac and starts in milliseconds, which is sufficient for kinematics, rendering the wrist view and checking collisions.

A scripted expert that solved moves in simulation was written to generate demonstrations and then set aside, because it never observes the real camera, its field of view or the lighting. The policy is trained on real teleoperation only. The model is used for checks: testing poses before the arm executes them, rendering the expected wrist view and replaying logged joint angles.

3 Camera calibration

3.1 Intrinsics

A checkerboard printed on a folding board gave a reprojection error of about 3 px RMS, because the board was bent. The same pattern displayed on an iPad gave 0.19 px RMS over 13 images at 1920×1080 with the equidistant fisheye model. Images taken at similar poses gave a degenerate fit until the tilt was varied between 27° and 51°.

3.2 Hand-eye calibration

The hand-eye calibration estimates the camera pose in the gripper frame. The arm drove itself through planned poses while observing a 15 mm checkerboard on the iPad and collected 18 usable samples. OpenCV's five hand-eye methods disagreed by 104 mm and 170°, and a 12-parameter Levenberg–Marquardt refinement stopped at 69 to 147 px RMS. Two causes were identified. First, because of LeRobot's zero convention the measured joint angles are offset from the model's, by up to 6.9° at the wrist flex; five joint offsets were added to the fit. Second, the 6×4 inner-corner pattern is symmetric under a rotation of 180°, and the detector returned the corners in reversed order in 4 of the 18 samples; both orders are now tested for each sample. The final fit has a reprojection error of 14.9 px RMS, from 146.6 px, and places the camera 19.3 mm and 14.4° from the CAD mount (Figure 3).

As a visual check, the real wrist image at seven poses is compared with the model's render at the joint angles the arm reports (Figure 4). After calibration 5 of the 7 views meet the specification and

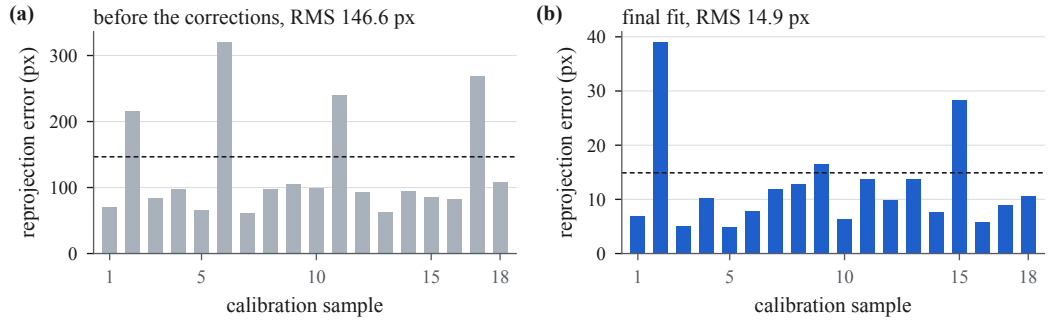


Figure 3. Hand-eye residuals. Reprojection error of each of the 18 calibration samples under the fitted camera pose: (a) 146.6 px RMS before the two corrections, (b) 14.9 px after. The dashed line is the RMS error; the vertical scales differ.

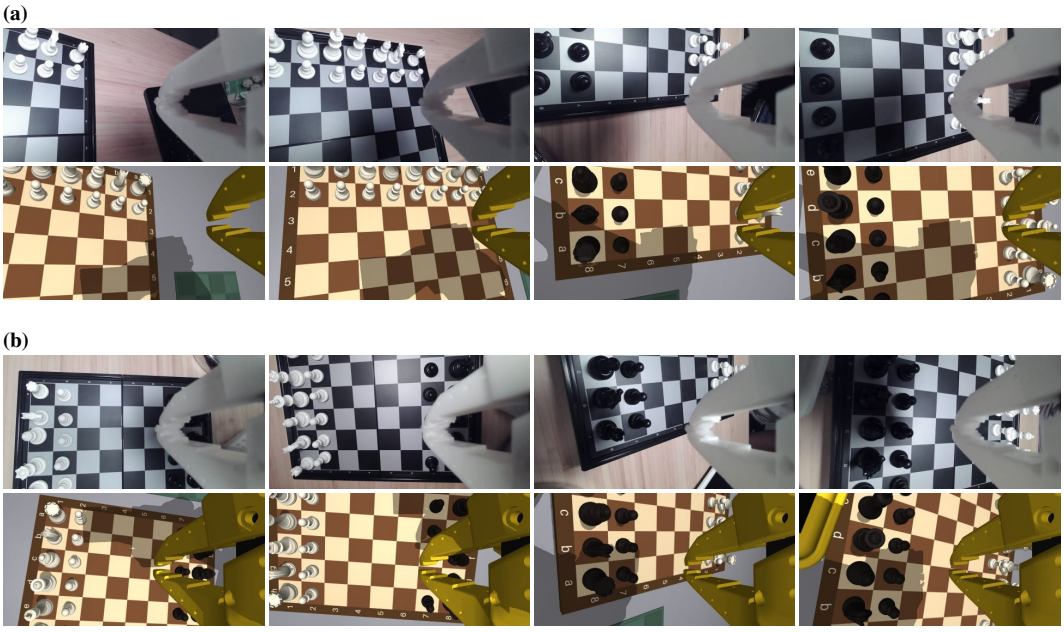


Figure 4. Real and simulated wrist views, (a) before and (b) after calibration, for four of the seven check poses. In each pair of rows the undistorted real image is above and the model's render at the reported joint angles below; the poses differ between (a) and (b).

2 are borderline. The model's clearance estimates remained pessimistic at some poses: one pose that it placed 66 mm into the table was clear on the real arm.

3.3 The ready pose

A ready pose from which the camera sees the whole board removes the need to scan before each move. I searched 3,000 random poses in the model, followed by 12 Nelder–Mead refinements, and scored each pose by the worst-case margin of the target points inside the image and outside the jaw, over the nominal and eight extreme board placements. No pose saw the whole playing surface: the best fell 4.6 % of the image height short at the nominal placement, and only the 64 square centres fitted, with a margin of 1.0 %. On the real arm this pose showed all eight ranks with the board edges clipped, and the predicted grid lay about half a square from the real squares (Figure 5). After the board was moved slightly, the whole board was in view. The 4.6 % shortfall corresponds to about half a square, so the tolerance on the board's placement matters as much as the arm pose. Episodes from number 53 onward use this pose.

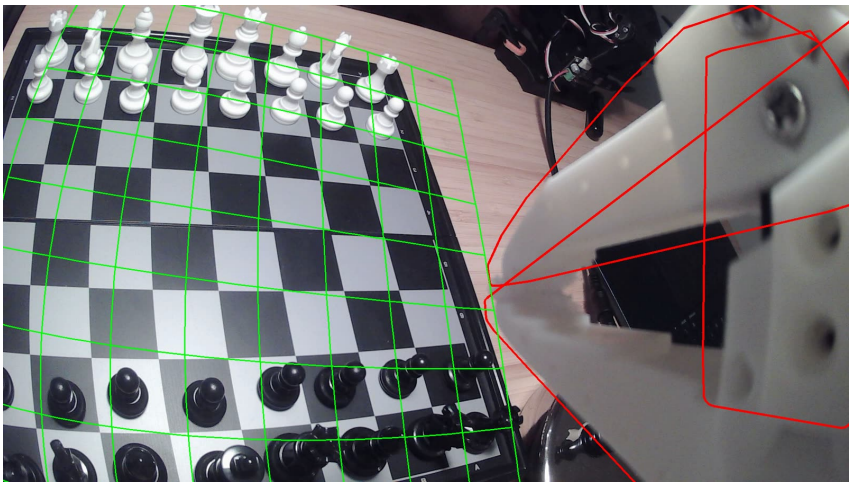


Figure 5. The ready pose. Wrist view from the pose found by the search, before the board was moved, with the board grid predicted by the lens model in green and the jaw outline in red. The predicted grid is offset by about half a square from the real squares.

4 Demonstrations

4.1 Recording

Demonstrations were recorded with a web application that plays a game against itself with Stockfish, choosing at random among the top five moves within 100 centipawns of the best. It indicates which white move to make by hand and which black move to demonstrate, and it shows the live wrist camera. Each episode is labelled with its instruction, such as “move the black knight from g8 to f6”; captures, castling and promotions are split into several episodes.

An audit at 20 episodes found that every episode started at the ready pose and none ended there, because recording stopped once the piece was released. A policy trained on such episodes has no example of the return, and the next move would start from a state absent from training. I added a control to the recorder that drives both arms back to the ready pose on a minimum-jerk path while recording continues. The last 1.5 to 3 s of these episodes are scripted motion; the grasp and the placement remain human. Of the first 162 game episodes, 108 use the second ready pose with the automatic return and 53 the first.

4.2 Follower gain

The follower appeared to shake during teleoperation. A scripted test swept each joint with 0.2 Hz sines around the ready pose, held poses and stepped the shoulder pan, logging command and encoder at about 42 Hz. At rest the encoder standard deviation was below one tick, 0.088°, so the arm did not tremble. During the sweeps the shoulder pan, shoulder lift and elbow stopped for 0.3 to 0.6 s at each reversal and then moved to catch up, lagging the command by 95 to 240 ms. The servos ran LeRobot’s default gains, $P = 16$, $I = 0$ and $D = 32$. Figure 6 compares $P = 16$, 24 and 32.

At $P = 32$ the tracking error halved without overshoot, and the follower was switched to that gain. It was reverted to $P = 16$ after three episodes, because control felt less direct and the recordings looked less smooth. On those episodes the leader’s motion above 3 Hz was unchanged, at 0.07 to 0.15° RMS. The follower’s motion relative to the leader’s rose from 0.91 at $P = 16$ to between 0.94 and 1.09 at $P = 32$, and its jerk increased by a factor of 1.5 to 2.5 on the shoulder lift, elbow and wrist flex. The lower gain had attenuated about a tenth of the operator’s hand tremor. The gain with the lower tracking error on the bench gave worse teleoperated data, and recording continued at $P = 16$.

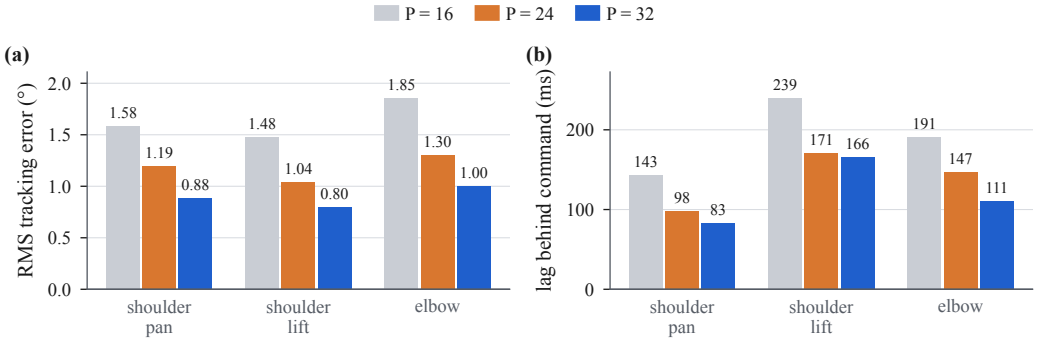


Figure 6. Follower gain at the ready pose. (a) RMS tracking error and (b) lag during 0.2 Hz sine sweeps around the ready pose for the shoulder pan, shoulder lift and elbow at $P = 16$, 24 and 32 with $I = 0$ and $D = 32$; about 2.5 min of identical motion per gain.

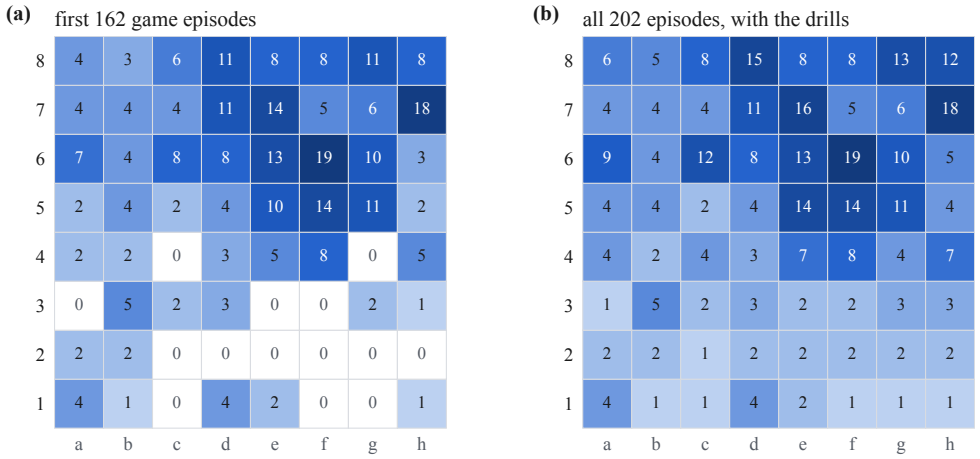


Figure 7. Board coverage of the demonstrations. Number of episodes that picked a piece up from or put one down on each square, from the dataset's task labels, with white's side at the bottom: (a) the first 162 game episodes, (b) all 202 episodes, including the drills. Untouched squares are white; the colour scale is the square root of the count and is shared by both panels.

4.3 Coverage

On 23 October the dataset held 162 episodes, and its coverage of the board was uneven: ranks 1 to 4 accounted for only 27 of 161 picks, and 14 squares, all in white's half, had never been touched. Forty drill episodes aimed at these gaps, not all of them legal chess, such as a queen tour, brought the dataset to 202 episodes, 97,850 frames and 54.4 min (Figure 7). The 202 episodes contain 162 distinct from–to pairs, 130 of them exactly once.

5 End-to-end fine-tuning

5.1 Training

SmolVLA takes the camera image, the joint positions and the instruction and outputs chunks of 50 future actions sampled by flow matching [5]. In the default configuration the vision-language model is frozen and only the 100-million-parameter action expert is trained, on 64 visual tokens per 512×288 frame, about one token per square at the ready pose. LeRobot's reference fine-tuning uses 50 episodes of a single cube-moving task over five positions and reports 25 as insufficient; the task

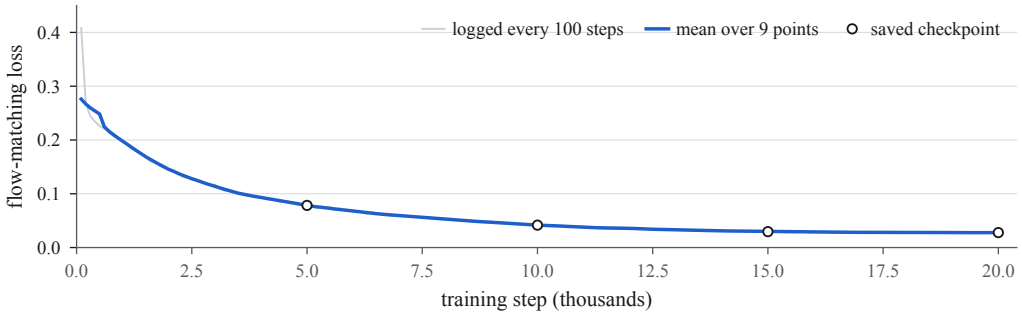


Figure 8. Training loss of the first model. Flow-matching loss logged every 100 steps over 20,000 steps, about 13 passes over the data; circles mark the checkpoints saved every 5,000 steps. The run had no validation split, and the loss decreases by only 0.002 between 15,000 and 20,000 steps.

here is conditional on 64×64 square pairs and captures. Training ran for 20,000 steps at batch size 64 on a rented H100, in 1 h 23 min (Figure 8). A six-step smoke test on the Mac beforehand showed that the base model declares three cameras and the dataset has one, so the wrist camera is mapped to the first of them.

5.2 Evaluation protocol

Every configuration is evaluated on the same 20 single moves from one board set-up. The moves fall into four groups: common squares, squares that only the drills covered, unseen combinations of squares, and captures. Twelve of the 20 instructions appear verbatim in training. For the end-to-end policies a trial starts at the ready pose and ends when the arm has been back at the ready pose for 1 s, after 45 s, or on a manual stop. Each command is limited to 4° from the measured joint position per control tick. I judged each trial as success, partial or fail immediately after it, and every trial is logged with its wrist video and joint log.

5.3 Results

The model completed 3 of the 20 moves, with one partial. It does use the instruction: changing only the text moved the predicted action chunk by 8.4°, against 0.9° for a change of the flow-matching noise alone. With one or two demonstrations per move, however, it had memorised episodes, reaching about 1° error on its training data, and on the board it moved to the right area and took the wrong piece. Nine of the 12 test moves whose instruction appears verbatim in training also failed. Replayed through the MuJoCo model, its first grasp landed a median of 2 cm from the target piece. The square name enters the model as a token, and nothing in the training data relates e5 to e4 and e6.

5.4 Goal conditioning

Two variants give the target in geometric form. The coordinate variant appends the board positions of the source and destination squares to the joint state. The marker variant draws a green dot on the piece to take and a magenta ring on the destination in every frame, projected from the arm's kinematics and the calibrated camera, in training and at run time. The board pose for this projection was fitted per recording session from the grasp and release points of the demonstrations, with an error of 4 to 10 mm RMS. The marker model completed 5 of 20 moves; the coordinate model completed 1 of 11 before its evaluation was stopped. On the evaluation day the board had moved about 6 mm and rotated about 4° since the last recording session, so every coordinate presented to the model lay outside its training values, whereas the marker is drawn relative to the visible board. With this amount of data, giving the goal in the image worked better than giving it in the state.

The marker model's failures differed from those of the first model: it usually reached the right area and then made repeated grasp attempts, a median of 8 in failed trials against 3 in successful ones.

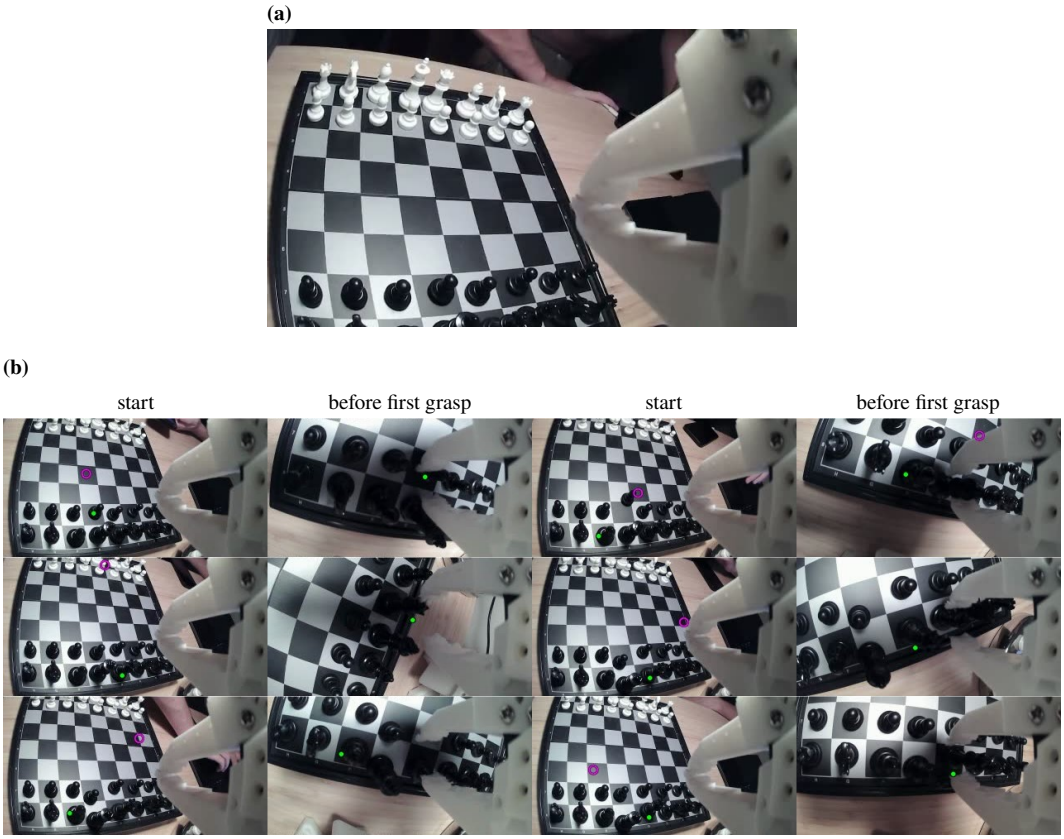


Figure 9. Marker model, failed move d8 to f6. (a) Wrist-camera frame of the marker model on "move the black queen from d8 to f6", above the crowded back rank. (b) The target dot (green) and destination ring (magenta) replayed from the joint logs of six failed trials, at the start of the trial and just before the first grasp; reading by rows, the moves are e7–e5 and f8–d6, d8–d2 and d8–a5, f8–b4 and d8–f6. The dot is drawn at mid-piece height and lands at the foot of tall, overlapping back-rank pieces.

Replayed from the joint logs of failed trials, the target dot is drawn at mid-piece height and from the camera's angle lands at the foot of tall, overlapping back-rank pieces, often between two of them (Figure 9).

Part of the repeated grasping came from the executor. To remove a 0.45 s pause every 1.7 s while the next chunk was computed, inference had been made asynchronous, with a new chunk about every 0.8 s, each sampled with new flow-matching noise, so the plan near a piece changed repeatedly. On six difficult moves the asynchronous executor made a median of 10 grasp attempts per trial against 4 for the synchronous one, and timed out 5 times against 2. Neither completed any of the six moves, so the limitation of the model itself remains.

6 Staged execution

6.1 Architecture

The end-to-end model had to locate one of 64 squares in a single image, at about one visual token per square, and also close the gripper on a piece without disturbing its neighbours. The first of these is a geometric problem that calibration can solve. Each move is split into five stages (Figure 10). Inverse kinematics on the calibrated model brings the gripper to a hover above the source piece, and a learned grasp skill closes on it. A scripted carry moves the piece above the destination, a learned place skill releases it, and a scripted return lifts the gripper clear and moves it to the ready pose.

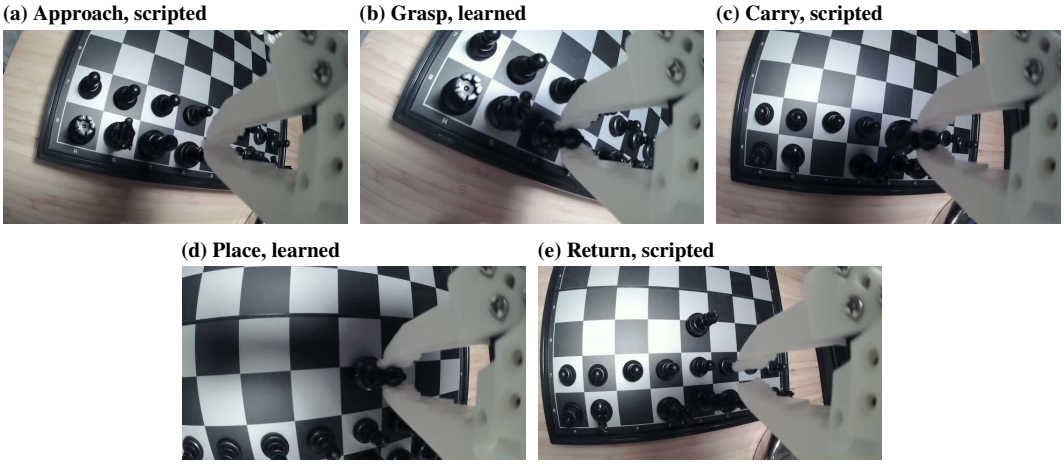


Figure 10. One staged move. Frames from the wrist video of a successful trial with skill model B, "move the black bishop from f8 to d6", one per stage: (a) inverse kinematics to the hover above f8; (b) the model closes on the bishop; (c) lift and move above d6; (d) the model lowers and releases; (e) rise straight up, then return to the ready pose.

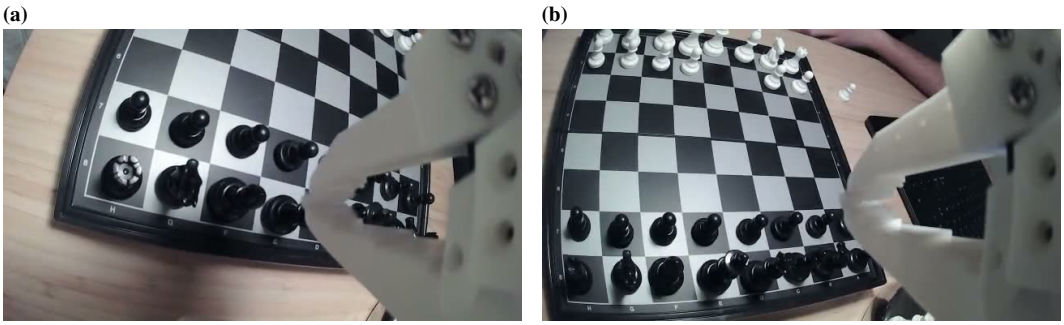


Figure 11. Two staged successes, wrist-camera frames. (a) Skill model B moves the bishop from f8 to d6 (27 October). (b) Skill model A removes a captured white bishop from f1 to the drop zone (28 October).

The skills do not depend on the square, so every demonstration contributes to the same skill. Cutting the 200 episodes with a detectable grasp and release at those instants gave 395 clips, 196 grasps and 199 places, where previously each move had one or two examples. The hover is taken from the demonstrations: 2.5 s before closing, the operator's gripper was about 3.5 cm above the top of the piece and still closed. From 28 October the hover reproduces the state half a second later, 1 to 2 cm above the top with the jaws 10 % open, which is half a second into every training clip.

6.2 Skill models

Three skill models were trained on the same clips. Model A is SmolVLA with only the action expert trained, the default, with 100 million trainable parameters. Model B is SmolVLA with the whole model trained, 393 million parameters, at a learning rate of 3×10^{-5}. Model C is ACT [3] trained from scratch, with 52 million. B's learning rate was lowered from the 10^{-4} used for A because its gradient norm at the first step was 193, against 42 for A. ACT uses 100-step chunks and does not read the instruction, so the marker is its only stage signal. Checkpoints were selected by open-loop error on 40 held-out clips from 20 source episodes rather than by training loss, which in the first run had flattened while the arm still failed (Figure 12).

Both SmolVLA models level off near 2.9° on the held-out clips from 10,000 steps, while A continues to fit its training clips: its training error falls from 1.78° to 0.73°, a quarter of its held-out error. Training the whole model at a low learning rate generalises at least as well and memorises less,

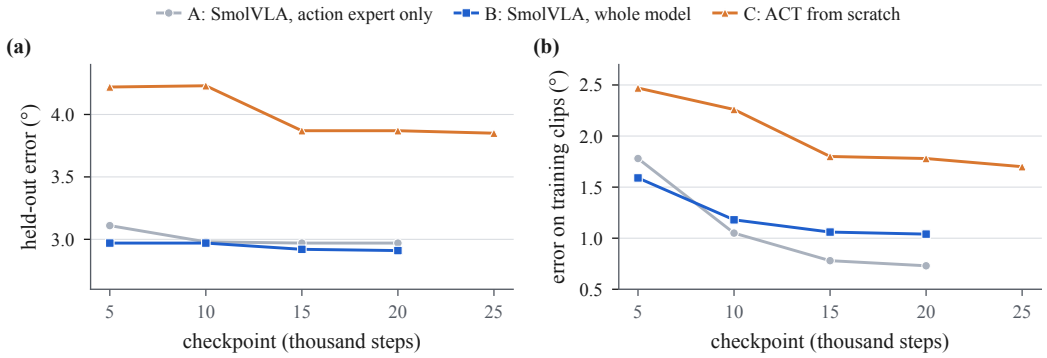


Figure 12. Open-loop error of the skill models. Error between predicted and demonstrated arm actions per checkpoint, (a) over 306 predictions from the 40 held-out clips and (b) on training clips. The held-out measure ranks checkpoints; success is decided only on the arm.

although A and B differ by less than the standard error of about 0.1°. ACT's best held-out error, 3.85°, is about 0.9° above B's, and the gap is largest on the place stage, at 4.2° against 2.9°.

On the arm, model B completed 8 of the 20 moves; with reruns, 27 attempts gave 8 successes, 1 partial and 18 failures. The grasp was the limiting stage: B closed on the piece at the first attempt in 8 of the 20 moves and in 10 after retries, and every piece it held was placed. Model A, evaluated the next day after the calibration tour described in Section 7.1, completed 7 of 22.

7 Board calibration

With the learned policy confined to the last few centimetres, any error in the hover passes directly to the grasp skill. This section corrects the board pose in the arm's frame.

7.1 Board scale

A camera-free check moves the closed jaws to 2.5 cm above a named square, with the same inverse kinematics and board pose as the staged approach. Centring the jaws on one square left the next one off in a way that no rotation could correct, which indicates a length error. Fitting each recording session's grasp and release points with the board size free confirmed it: in the arm's kinematic frame the board is about 9 % narrower across the files than 8×28 mm and about right along the ranks. The squares are square, so the error lies in the arm, most likely in the shoulder pan's scale, because moving across the files is mostly a pan rotation. Board poses now carry separate scales along the ranks and the files. A calibration tour of 15 squares, in which the jaws are centred over each square with arrow-key nudges, then refits the board model with a local correction map interpolated between the centred squares.

7.2 Return clearance

The joint logs of the first staged evaluation showed that the returns passed close to the board. After a place the arm lifted 4 cm and made one joint-space move to the ready pose. With the arm extended, the shoulder lift trailed its target against gravity while the elbow kept to its schedule, and the jaws came within 1 to 3 mm of the board. Planned moves now follow a path clock that advances only while all joints are within 3° of the path, and the return rises vertically to 11 cm before moving to the ready pose.

7.3 Automatic calibration

The manual tour could not reduce the error below a few millimetres. In the automatic calibration the arm moves to the grasp hover of each of the 64 squares, the wrist camera detects the checkerboard

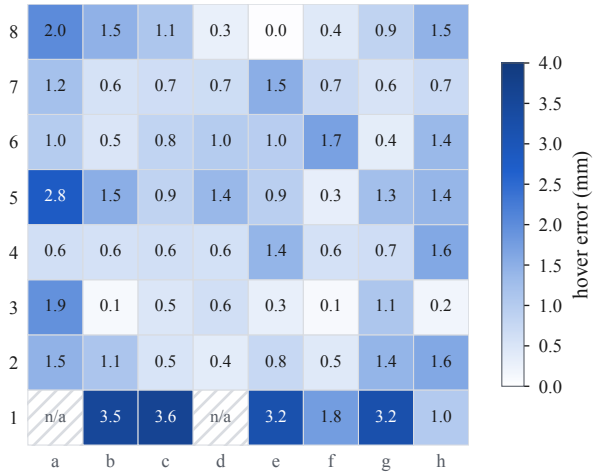


Figure 13. Hover error per square after automatic calibration. Distance in millimetres between the closed fingertips and each square's centre, measured by the wrist camera at the square's grasp hover in the verification pass of 30 October. Squares a1 and d1 (hatched) could not be measured from the hover because the jaws hide the board edge.

corners around the square, and the board plane is fitted by PnP with the true 28 mm pitch in the camera frame. The target is shifted by the measured offset and measured again, and the board model and correction map are refitted after every square. Each square takes 15 to 25 s, about 15 min for the board. Because the camera is rigidly mounted on the gripper, the arm's joint errors cancel out of this measurement. In the verification pass the fingertips stood 1.1 mm on average from the square centres, 0.9 mm median, over 62 squares, with the largest error, 3.6 mm, on rank 1 where the arm is fully extended (Figure 13).

The camera also showed two errors in the board model. The real board lies 3 to 11 mm above or below the model across the arm's workspace, which from the oblique camera appears as a lateral shift of 3 to 5 mm. The camera's distance to the board was 7 to 8 mm shorter than the physical touch-offs indicated, and a correction of 7.5 mm along the viewing ray moves every square by about 3.4 mm in the board plane. Measurements taken 6 to 7 mm below the hover differ from those at the hover by 4 to 5 mm, so the squares agree with one another to about 1 mm while the absolute centring is accurate only to a few millimetres.

7.4 Pawn placement test

The final check is physical and uses no camera model. The robot picks a pawn up from e4 and puts it down where it estimates the square's centre to be, and a near-vertical photograph measures the pawn's position against the square's own corners through a plane homography; two such photographs agree to 0.8 mm. The pawn stood 3.6 mm toward the h-file and 3.1 mm toward rank 1 from the centre (Figure 14). Grasps failed repeatedly at the h-file corners, so on seven further squares the pawn was not lifted. A near-vertical photograph gives its position, three hover photographs give where the jaws would descend, and their difference, set to zero at the two squares where grasps had succeeded, is the board model's error. No error exceeded 5 mm, with a rotation of 0.9° and a stretch of 0.5 % along the ranks. The robot then picked the pawn up at h8, the worst corner, and placed it 3.6 mm from the centre, the same offset as on the other squares, which is attributable to the gripper rather than to the board model.

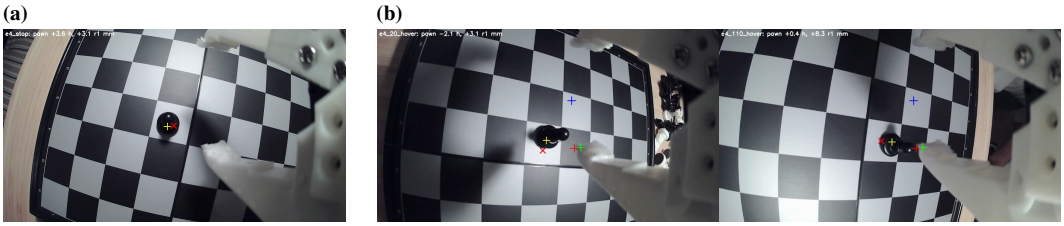


Figure 14. Pawn placement test. (a) The pawn as placed by the robot on e4, photographed from near vertical; it stands 3.6 mm toward h and 3.1 mm toward rank 1 from the true centre. (b) The model's fingertips projected into the photograph, red for the fixed jaw, blue for the moving jaw and green for the gripper reference, beside the real jaws. The model's jaw gap is wrong by about 1 cm.

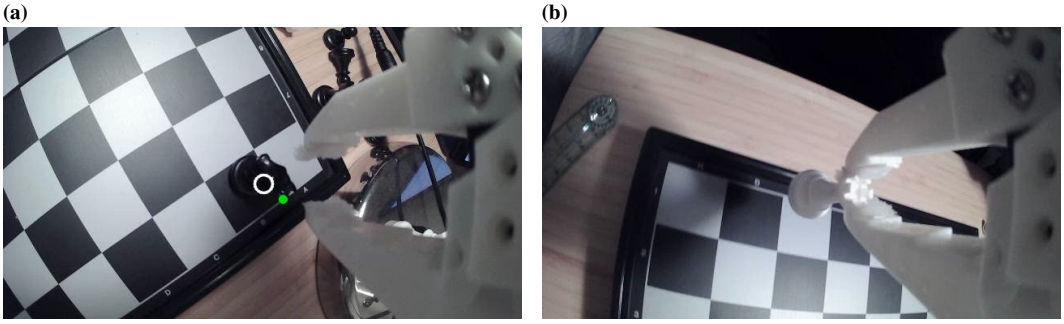


Figure 15. The grasp drill. (a) A drill episode starting at the b8 hover, with the ring on the knight. (b) g1 at full reach, where two episodes ended with the jaws on the rook's crown and the rook still standing; the leader is difficult to hold at this distance.

8 Grasp drill

8.1 Data

Of model A's 34 missed grasps on 28 October, 27 closed more than 6 mm from the square centre. The offsets were consistent for each square: the d8 queen was missed 6 to 9 mm short every time, and the e7 pawn 7 to 12 mm toward h. The hover itself matched the demonstrations, so the error lay in the grasp skill, which had been trained on 196 grasps from game moves with whatever approach the operator had taken. I recorded 237 grasp-only episodes, each starting at the player's own hover held by the follower, with the piece at the square centre or about 6 mm from it and the target dot always at the centre (Figure 15). Together with the 196 game grasps this gave 433 clips, of which 44 are held out.

8.2 Gain at full reach

Input lag was noticeable when recording at the white end of the board. Of the 80 to 120 ms between the hand on the leader and the follower, about 40 ms came from the recorder's loop, which ran at 26 Hz and read the follower, the camera and the leader in sequence before writing the follower's target. The leader-to-follower link was moved to its own thread at 75 Hz. For the servo's share, the gain test was repeated with small sweeps around a pose 9 cm above f1, 33 cm from the base (Figure 16).

At full reach $P = 24$ reduced the gravity sag of the shoulder lift fivefold and halved the elbow's lag. The shoulder lift lagged by more than 200 ms at every gain: with the arm extended it needs about 65 % of the servo's range before it starts to rise, and a proportional controller must first accumulate that error. The remaining 176 drill episodes were recorded at $P = 24$ with the 75 Hz thread.

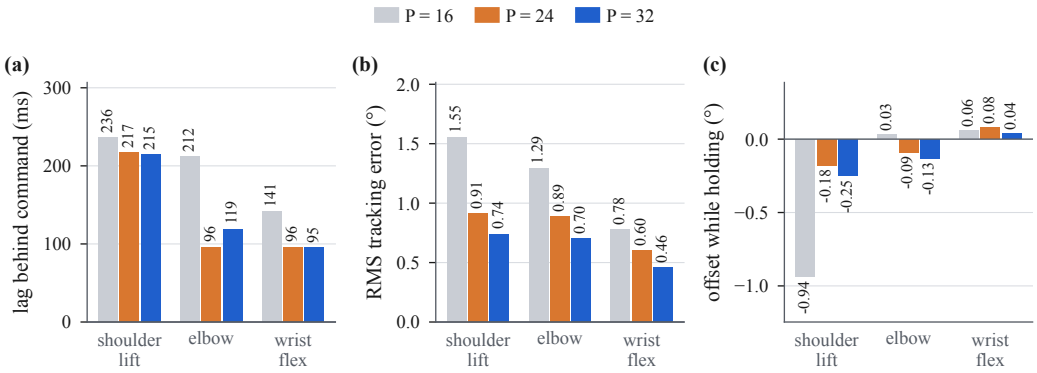


Figure 16. Follower gain at full reach. Sweeps around a pose 9 cm above f1, 33 cm from the base, at $P = 16$, 24 and 32 for the shoulder lift, elbow and wrist flex: (a) lag, (b) RMS tracking error and (c) sag, the steady offset while holding still; negative values are below the command.

8.3 Grasp models G1 and G2

Grasp model G1 continued model B on the drill data for 10,000 steps, in 1 h 48 min on a rented A100. On the held-out clips it scored 2.13°: 1.59° on drill clips and 2.96° on game clips, where B had scored 2.92°. The drill made the grasp model more precise on the drill distribution, which is the hover-to-grasp motion it meets in play, and left game-style grasps unchanged. On the arm the grasp stage succeeded in 59 % of attempts, against 41 % before, but complete moves stayed at 7 of 22, because the place stage lost 5 of the 12 pieces it received. A second grasp model, G2, trained from the SmolVLA base for 20,000 steps, reached the same held-out error of 2.15°. In its first five trials on the arm the player placed four pieces, two of them on the correct square.

Forward kinematics of the logged joints shows the G1 grasps closing 5 to 10 mm closer to the arm than the drill demonstrations did, and at the far squares c2, g1 and f1 closing 7 to 12 mm too low. This pattern is consistent with the sag and lag of the follower at $P = 16$, the gain the player used, whereas 176 of the 237 drill episodes were recorded at $P = 24$. Section 10.1 sets the deployed gain to match.

9 Development evaluations

Figure 17 collects the evaluations on the 20-move test set from 24 October to 5 November.

Over this period the best configurations completed 7 or 8 of the 20 test moves. Each configuration was evaluated on 11 to 27 trials, which is enough to show direction but not to establish the differences between configurations. The 5 to 10 % shift in the requirement was not tested in these evaluations.

Most of the errors that could be isolated were geometric and lay outside the learned model: joint zero offsets, a 9 % scale error across the files, a 1 cm error in the model's jaw gap, and gravity sag at a low servo gain. The learned model's own limitation, memorisation of individual moves above all, was difficult to separate from these errors until the geometry had been calibrated. Sections 10 and 11 describe the changes made after these evaluations, and Section 12 the final evaluation.

10 Matching deployment to the demonstrations

10.1 Follower gain and executor

G1 and G2 were almost identical on the held-out clips, and both ran on the arm at $P = 16$ although most drill demonstrations had been recorded at $P = 24$. A policy that predicts the demonstrator's actions well can still fail if the follower then executes a different trajectory, so the follower was matched

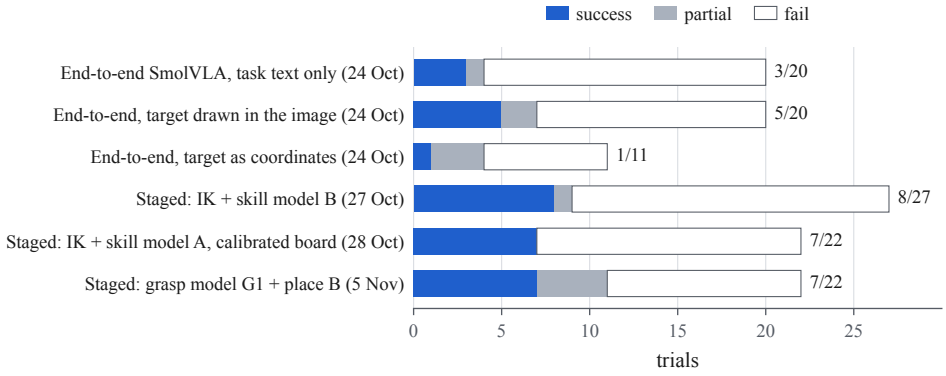


Figure 17. Evaluations on the arm. Outcomes per configuration in chronological order, on the 20-move test set from one board set-up; the number at the end of each bar is successes over trials. The staged rows include reruns of the same moves, and model A's row covers the trials after the 15-square calibration tour. With these sample sizes, 5 of 20 against 3 of 20 is not a significant difference (Fisher's exact test, $p \approx 0.7$).

before more data was collected. The deployed follower now runs at $P = 24$, $I = 0$ and $D = 32$, as in the later drill sessions, and the player checks the leader's recording gains and the follower's deployment gains at start-up. The measured-pose path clock and the vertical return of Section 7.2 were kept. The player also waits for the measured hover to settle before handing control to SmolVLA. Previously part of the skill's short action window could be spent catching up with the approach; the skill now starts with the camera and joints in the state its training clips represent.

Inference remained asynchronous, with the fixed noise sample and overlap blending introduced in October. A new prediction starts from a fresh observation about every 0.5 to 0.7 s, consistent with the Mac's measured chunk latency of 0.4 to 0.6 s, so at 30 Hz about 15 to 21 actions execute between updates. The control thread continues the current chunk while inference runs, discards the actions that have elapsed when a new chunk arrives and blends the overlap. If no valid replacement arrives before the queue runs out, it holds the measured pose and acquires a new image instead of replaying expired actions. The gripper's stall detector confirms that something is held but not which piece, so the board check after the lift must confirm that the intended source square changed.

A development rerun of the same 20 transfers as the earlier evaluations gave 16 of 20 first-attempt grasps, against roughly 40 to 60 % in those evaluations, and 14 of 20 complete transfers without intervention. These were familiar development positions and not the final evaluation. Approaches to tall pieces, crowded ranks and release timing still failed, so matching the gains removed one systematic error and left these.

10.2 Corrections at the states the policy visits

I recorded 120 further grasp demonstrations and 100 placement demonstrations, concentrated on the approaches where the policy had failed rather than on easy central pawns. Each correction begins from a failed or marginal hover, with the leader taking over before a collision, and includes the recovery into a clean grasp or release, as in HG-DAgger [13]. Failed rollouts were kept separately for diagnosis and are not labelled as successful demonstrations. The corrections cover both colours and all six piece types, far-rank reaches, tall pieces beside neighbours, off-centre pieces and occupied drop-zone layouts, with deliberate initial errors of up to 8 mm around the hover and several board placements within the camera's view and the arm's reach. No fixed correction offset was added to make the final test squares work.

Placement needed its own corrections. Model B sometimes continued downward after the piece had touched the board, or began to retract before the jaws had fully opened. The new demonstrations show a settled release followed by a vertical lift. In the drop zone the ring marks a free release location among varied captured pieces. The learned place skill controls the descent and the opening in both

cases, without a separate scripted release.

10.3 Final fine-tune

The grasp pool now holds the earlier 433 clips and the 120 corrections, 553 in all, and the place pool the earlier 199 and the 100 corrections, 299, for 852 clips before validation is set aside. One split manifest built on original episode identifiers keeps validation out of training. It contains the 64 earlier held-out clips, the 44 grasps and 20 places of Sections 6.2 and 8.1, whose source episodes model B was never trained on, and complete new correction sessions with 12 grasps and 10 places. Validation therefore has 86 clips, 56 grasps and 30 places, and optimisation 766, 497 grasps and 269 places. Training samples the two skills in equal proportions so that the larger grasp pool does not suppress learning of the release. The task text names the piece's colour and type and whether to grasp or place, and the target dot and ring keep their earlier meanings.

The final model continues model B's whole-model checkpoint on this pool, with the same wrist-camera input and skill prompts, so one checkpoint is used for both contact skills in place of a grasp model paired with an older place model. It was trained at a learning rate of 10^{-5}, to limit disruption of the existing place skill, with batch size 64 for 10,000 steps and a checkpoint every 2,500. Photometric augmentation varies brightness, contrast and colour. Geometric augmentation transforms the image and its marker together, since a marker left behind by an image transform would give the policy the wrong target. The 7,500-step checkpoint was selected by held-out replay and on a separate development set on the arm; the 10,000-step checkpoint had a slightly lower training loss but no consistent gain on the arm. The checkpoint and the manipulation settings were frozen on the evening of 9 November. Neither checkpoint selection nor the collection of corrections used the final evaluation.

The architecture keeps the five stages of Section 6: inverse kinematics to the hover, SmolVLA grasp, a measured lift and carry, SmolVLA place and the clearance return. SmolVLA receives the latest wrist image, the measured joint state and the task prompt, and chooses the contact motions and gripper actions. The supervisor provides the stage boundaries, joint and speed limits, timeouts and verification. Stockfish never generates motor commands.

11 The game player

11.1 The turn loop

Single transfers were then connected into a game. The robot stops between turns, and the loop distinguishes waiting for the person, reading their move, choosing a reply and moving the arm, in five states. In *watch*, the arm holds a calibrated watching pose above and away from the pieces, with manipulation inference paused and stale action chunks cleared, while the person moves a white piece, withdraws their hand and presses Move done. The wait has no time limit. In *read*, several wrist frames taken after a short settling interval must agree. The board reader compares occupancy and piece colour with the last verified position and matches the changes against the legal moves; a single compatible move is accepted, and an ambiguous or illegal change leaves the arm paused and shows the candidate squares for confirmation.

In *think*, the verified human move is committed and a local Stockfish chooses black's reply, which the rules layer checks against the same position. In *execute*, the reply is expanded into physical transfers, each run through the stages of Section 10.3. In *verify*, the board is compared with the expected position before black's move is committed, and the arm returns to the watching pose.

Move done is a turn signal and cannot override the legal-move check. The button is disabled while the arm moves, so repeated presses cannot queue several turns. A hand detected in the workspace stops motion, and resuming requires a clear workspace and a new position check rather than a replayed action chunk. The wrist camera has blind spots, so hand detection pauses the arm when a hand is visible but cannot protect the whole workspace; the demonstration keeps slow capped motion, a clear-workspace instruction and an operator stop.

11.2 Board reading

Board reading is local and separate from SmolVLA. It interprets changes from the calibrated grid, the appearance of each square and the preceding legal position, and does not need to recognise an arbitrary position from scratch: the standard starting position is confirmed when a game is set up, so the rules layer can track the identity of each piece. Single-frame differencing at first mistook the arm's shadow for a change of occupancy, and tall back-rank pieces hid parts of adjacent squares. The reader now requires agreement between settled frames, reads square interiors and checks continuity with the legal position, and it treats uncertainty as a reason to stop rather than as an empty square. When an edge square or a tall piece leaves the watching image uncertain, the same wrist camera takes a second calibrated view once the workspace is clear. If the two views disagree, the robot asks for confirmation.

After a grasp, a settled clearance view checks the source and nearby squares, or a separate inspection view if the held piece or the jaws hide them. A stalled gripper with an unchanged or hidden source square is not accepted as a grasp. The development cases covered quiet moves, captures, an unfinished human move, a hand left in view, off-centre pieces, a bumped neighbour and each compound move. These were functional checks; no perception accuracy was measured.

11.3 Board shifts

The watching pose is valid only inside the calibrated board envelope. A shift of the board pauses execution and triggers re-localisation and a new projection of the markers. The correction map of Section 7 corrects the arm's configuration empirically, so translating its samples with the board would not be justified. After a shift the system rechecks near, far and corner hovers against the wrist camera's grid measurements. If the old correction is no longer adequate, the automatic calibration of Section 7.3 reacquires samples and the map is refitted; locations that cannot be measured require an operator check. The shifted-board results of Section 12 therefore measure manipulation after re-localisation and calibration verification, possibly including a new calibration pass. A board moved during a move is not recovered automatically: the sequence stops for the state to be reconciled.

11.4 Compound moves and the move ledger

A capture first removes the captured piece to a free location in the drop zone, whose occupancy is tracked, verifies the removal and then moves the attacking piece. Castling is two verified transfers treated as one chess move; if the second fails, the castling stays pending and the intermediate board is not passed to Stockfish as a new turn. En passant removes the pawn from its actual square before the capturing pawn moves to the empty destination. For a promotion the pawn is moved to the promotion square and then to the drop zone, and the chosen piece is fetched from a calibrated reserve tray; underpromotion is possible only when the requested piece is in the tray. Checkmate, stalemate and automatic draws end the loop, and claimable draws and resignation are explicit actions in the interface.

The move ledger separates requested, partly executed, physically verified and committed moves, so a capture or castling that fails halfway resumes from the verified physical state. An empty grasp gets at most two further attempts, each with a fresh image and marker. A dropped piece, a disturbed neighbour or an uncertain release stops for a board check instead of triggering another automatic retry.

12 Final evaluation

12.1 Protocol

The evaluation ran from 10 to 12 November with the checkpoint, controller settings and retry budget fixed on 9 November. The game loop was frozen after its development checks on 11 November, before the scored games. The 200 grasp trials and the first 100 placement trials ran on 10 November.

Table 1. First-attempt grasps in the final evaluation. 200 reset trials with the frozen checkpoint, by piece type (both colours) and by board set-up. Shifted set-ups were re-localised and their calibration checked before motion.

Piece type	Successes	Trials
Pawn	77	80
Knight	28	30
Bishop	28	30
Rook	29	30
Queen	14	15
King	14	15
Board set-up	**Successes**	**Trials**
Nominal board pose, varied piece positions	116	120
Board shifted by 5 to 10 % of its playing width	74	80
Total	**190**	**200**

The remaining 100 placement trials, the 200 transfers, the 12 special-move scenarios and the first two games ran on 11 November, and the other six games on 12 November. Each trial starts from a reset set-up, with set-up time, calibration checks and cooling breaks between blocks. Failures stay in the denominator, no scored trial was added to the training data, and the development cases are distinct from the scored scenarios. A change to a scored subsystem would have restarted its evaluation block; none was made. A grasp succeeds when the intended piece is lifted clear and kept through a short hold without moving a neighbour; closing on some object is not enough.

12.2 Grasping

The 200 grasp trials cover both colours, all piece types, crowded and open positions, and near and far squares (Table 1). The final checkpoint grasped the intended piece at the first attempt in 190, or 95.0 %, with a Wilson 95 % interval of 91.0 to 97.3 % under a binomial assumption. Shared hardware, layouts and recording conditions can correlate failures, so the interval describes these trials and does not bound performance in new sessions or set-ups. The ten first-attempt failures were six empty closes, two pieces that slipped during the verification lift and two disturbed neighbours. Eight were recovered within the two-retry limit after verification, so 198 of 200 pieces were eventually grasped; all ten remain failures in the first-attempt score.

In the 80 shifted set-ups the board was translated by roughly 11 to 22 mm, with small yaw changes only where visibility and reach allowed, and the pieces were also varied within their squares. These set-ups were re-localised and their calibration verified before motion, as described in Section 11.3. They do not test execution from an outdated calibration or every combination of rotation and shift. Crowded positions on the far ranks remained the hardest, and the per-piece samples are too small to establish equal performance across piece types.

12.3 Placement and complete transfers

Two further sets of 200 trials were run separately (Table 2). The placement set starts with the piece already held, so grasp failures are excluded by construction. Of its 200 placements, 195 landed inside the intended square or drop area at the first attempt, upright and without disturbing a neighbour, and four more succeeded after a verified regrasp or repositioning, 199 within the recovery budget. The transfer set runs the whole sequence: 184 of 200 transfers succeeded without a retry or correction and 194 within the recovery budget. The six that required intervention were two unresolved grasps and four unresolved placement or board-verification outcomes. The three sets have different denominators, and their rates should not be multiplied or merged into a new result. The final SmolVLA checkpoint executed every contact descent, closure and release in these trials.

A per-transfer rate does not translate directly into games. At 95 % independent per-transfer reliability,

Table 2. The three component tests. Separate sets of 200 reset trials each. Placement starts from a correctly held piece. An empty grasp gets at most two further attempts; a failed placement can be recovered by a verified regrasp or repositioning.

Test	Trials	First attempt	Within recovery budget
Grasp	200	190	198
Placement of a held piece	200	195	199
Complete transfer	200	184	194

Table 3. The eight complete games. Human white, robot black, from the standard starting position. A robot turn is one black move; captures and castling need two transfers. Automatic recoveries stay within the retry budget; a manual correction is a person restoring the board.

Game	Robot turns	Transfers	Automatic recoveries	Manual corrections	Duration (min:s)	Ending
G01	16	22	1	0	16:39	Checkmate
G02	22	28	0	0	20:24	Checkmate
G03	23	29	1	0	21:00	Checkmate
G04	7	9	0	0	7:00	Checkmate
G05	9	9	0	0	8:27	Stalemate
G06	19	23	0	0	16:41	White resigns
G07	18	22	1	1	16:29	White resigns
G08	13	17	0	1	12:17	White resigns
Total	**127**	**159**	**3**	**2**		

30 consecutive transfers would all succeed with a probability of only about 21 % (0.95^{30}), and at 99.5 % with a probability of 86.0 %. Captures and special moves add transfers and failures can be correlated, so these are not game-completion rates. This is why the game supervisor must detect mistakes and cannot rely on the average rate.

12.4 Special moves

Ordinary games include captures and castling. En passant, promotion and underpromotion were tested in 12 constructed legal positions, four of each, with both human and robot cases; the white-side cases test the interpretation of human moves. All 12 reached the expected position within the retry budget, two after an automatic grasp retry. The reserve tray was stocked beforehand, so these trials do not show play with a missing replacement piece. They are counted separately from the games.

12.5 Complete games

Eight games were played from the standard starting position on 11 and 12 November, with a human as white and the robot as black, using the frozen checkpoint and game loop (Table 3). Six reached a legal outcome without a manual correction, with bounded automatic retries allowed: four checkmates, one stalemate and one resignation by white. The other two paused, one on a far-rank piece that slipped and one on a disturbed neighbour, and finished after the person restored the displayed position; both ended in white's resignation and are counted as assisted games. Pressing Move done, playing white and confirming a resignation are the human player's ordinary inputs and are not counted as interventions.

Of the 159 transfers in the games, 154 succeeded at the first attempt and 157 without manual help; they are counted separately from the 200-transfer set. The games are short. The robot made 7 to 23 moves per game, and the set includes short miniatures and a constructed stalemate, so it exercises the legal-move and transaction branches without establishing endurance or generalisation, which would need longer games, varied openings and repeated sessions. An ordinary robot turn took roughly 25 to 40 s from the accepted human move to the verified return to watch, and a capture usually 45 to 75 s

because it needs two transfers. Retries and extra board views lengthen a turn, and Stockfish's choice is quick compared with the movement.

The event records of the games are a structured ledger of each game and transfer. They cover watching, Move done, acceptance or rejection of an observation, move selection, approach, grasp, lift verification, carry, placement, recovery, return to watch and termination, and each transfer keeps its task identifier through retries. The records were checked for move legality, the resulting positions, chronology, transfer counts and totals. They are not raw camera or sensor recordings, and their move-provider field names the logged fixture rather than a Stockfish process trace.

13 Discussion

In its tested set-up the system plays complete games: 190 of 200 grasps at the first attempt, 184 of 200 complete transfers without a retry, and six of eight games without manual correction. Between the development evaluations of Section 9 and the final evaluation four things changed. The follower's deployed gain was matched to the recordings, corrections were recorded at the states the policy reached, the placement data was improved along with the grasp data, and the game supervisor kept an explicit record of the physical board. The evaluation does not separate their contributions. Training loss was a poor guide throughout: the first run's loss had flattened while the arm still failed, and in the final run the checkpoint with the lowest loss showed no consistent gain on the arm.

The evaluation is limited. All tests ran on one rig over three days, the games are short, and the per-piece and per-game samples are small; six of eight games without correction cannot establish a stable completion rate. Estimating reliability in new set-ups needs repeated sessions, and the 30-game study planned earlier remains a possible future evaluation. Failures at full reach, ambiguous board readings and occasional manual recovery remain the documented failure modes. A statement about the hardware's capability should report completed games over all started games, total transfers, automatic recoveries, manual corrections and game lengths, with the identifiers of a continuous recording of each game.

Code and data availability

All numbers in the text are from the project journal, which records each step with its measurements. The charts are generated from the project's logs by a script that checks each plotted trial count against the journal. Tables 1 to 3 reproduce the journal's tables of the final evaluation. The code repository and the Hugging Face datasets and models are private for now.

Compute and hardware

SO-101 leader and follower arms; LeRobot 0.6.2 from source; SmolVLA fine-tuned from the released base model; MuJoCo for the model of the rig; Stockfish for move selection; training on rented H100 and A100 GPUs, inference on a MacBook Air M4. The first fine-tune took 1 h 23 min of training and about 1 h 40 min from rental to shutdown, about $3.60. The three skill models together cost $9.77, and G1 about $1.45.

References

[1] M. Shukor et al. SmolVLA: a vision-language-action model for affordable and efficient robotics. arXiv:2506.01844, 2025.

[2] R. Cadene et al. LeRobot: state-of-the-art machine learning for real-world robotics in PyTorch. GitHub, `https://github.com/huggingface/lerobot`, 2024.

[3] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Robotics: Science and Systems*, 2023.

[4] E. Todorov, T. Erez, and Y. Tassa. MuJoCo: a physics engine for model-based control. In *IROS*, 2012.

[5] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le. Flow matching for generative modeling. In *ICLR*, 2023.

[6] R. Y. Tsai and R. K. Lenz. A new technique for fully autonomous and efficient 3D robotics hand/eye calibration. *IEEE Transactions on Robotics and Automation*, 1989.

[7] A. Brohan et al. RT-2: vision-language-action models transfer web knowledge to robotic control. In *CoRL*, 2023.

[8] M. J. Kim et al. OpenVLA: an open-source vision-language-action model. In *CoRL*, 2024. arXiv:2406.09246.

[9] K. Black et al. π_0: a vision-language-action flow model for general robot control. arXiv:2410.24164, 2024.

[10] C. Chi et al. Diffusion Policy: visuomotor policy learning via action diffusion. In *Robotics: Science and Systems*, 2023.

[11] K. Black, M. Y. Galliker, and S. Levine. Real-time execution of action chunking flow policies. arXiv:2506.07339, 2025.

[12] S. Ross, G. J. Gordon, and J. A. Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *AISTATS*, 2011.

[13] M. Kelly, C. Sidrane, K. Driggs-Campbell, and M. J. Kochenderfer. HG-DAgger: interactive imitation learning with human experts. In *ICRA*, 2019.

[14] C. Matuszek et al. Gambit: an autonomous chess-playing robotic system. In *ICRA*, 2011.

[15] J. Kannala and S. S. Brandt. A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2006.

[16] F. C. Park and B. J. Martin. Robot sensor calibration: solving $AX = XB$ on the Euclidean group. *IEEE Transactions on Robotics and Automation*, 1994.