THE REPAIR MUST REACH THE BACKBONE: A CONTROLLED COMPARISON OF INSTRUCTION-FOLLOWING FIXES FOR VISION–LANGUAGE–ACTION POLICIES

Anonymous authors
Paper under double-blind review

ABSTRACT

A vision–language–action policy can clear every task in its training suite and still reach for the object its instruction did not name. Repairs exist, and each one arrives on its own base policy, its own demonstrations and its own benchmark, so the field cannot say which ingredient does the work. We hold the base, the demonstrations and the protocol fixed, and compare sixteen arms on one ruler: change one word of the instruction, hold the scene and the robot pose, and record which object the arm reaches, against a random-motion floor. The base reaches the named object in 36% of trials. A swap-differential probe shows that its backbone already picks out the newly named object, so the natural repair is to teach the action expert to read what the backbone computes. That repair does nothing. Training the action expert alone, at either rank, leaves the named-object rate where the base had it, and on the decorrelated demonstrations it also cuts success on the trained tasks by more than half. Given only what the robot observes, no repair that leaves the vision–language backbone frozen passes 53%. Every repair that passes it from the robot's own observations reaches the backbone, and the two that lead decorrelate the instruction from the scene layout: counterfactual relabelling, which edits the word, and low-rank adaptation on position-randomized demonstrations, which moves the object. The second reaches 97.7% across three seeds. Activation patching locates that access: in the pairs the base itself tells apart, swapping the keys and values of six consecutive backbone layers flips its choice 29 times in 32, no other band flips more than once, and the repair keeps that band and makes it more reliable. The recipe then carries off the checkpoint it was tuned on. A second base rises from the same 36% to 77%, a third-party counterfactual benchmark shows the repaired policy following the instruction where the base replays its demonstration, and objects whose name and mesh never occur in training are selected at twice the chance rate. We release the policies, the demonstrations and their generator, and the measurement code.

1 INTRODUCTION

A vision–language–action (VLA) policy can finish its whole suite without reading a word of its instruction. The base policy we study completes the original LIBERO object tasks on 198 of 200 trials. Replace its instruction with the string `xxx`, change nothing else, and it still completes 89 of them, walking to the object that usually occupies the target position. Standard suites hide this. They place the named object where the demonstrations put it, so a policy that resolves the name and a policy that replays a position earn the same score (Zhou et al., 2025; Fei et al., 2025; Yu et al., 2026). Move the name and hold the object, and published scores collapse (Fang et al., 2026b; Chen et al., 2026). Behaviour cloning takes the cheapest feature that predicts the expert action, and a fixed target position is cheaper than a word (de Haan et al., 2019; Geirhos et al., 2020; Pezeshki et al., 2021). The literature calls the result instruction blindness (Zhang & Bisk, 2026).

Repairs exist, and they act at three different places in the policy. CAST rewrites the instruction in the demonstrations (Glossop et al., 2025), following a longer line of counterfactual relabelling

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

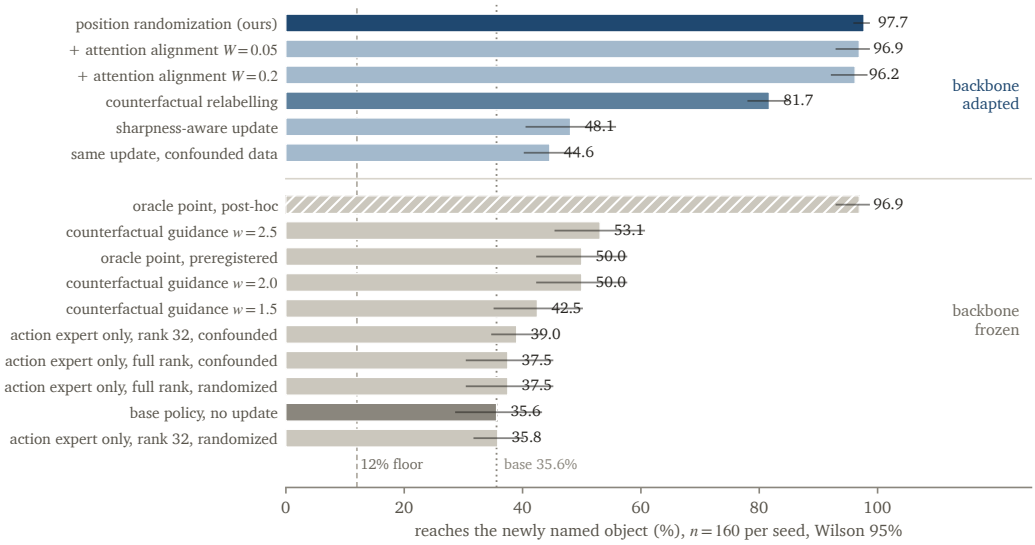


Figure 1: Sixteen arms on one base policy, one set of demonstrations and one ruler. Each bar is the rate at which the arm reaches the object the instruction names within 2 cm after one word changes and nothing else does (SWAP-OBEY; n=160 per seed, three seeds for the multi-entry arms, Wilson 95%). We measured every bar in this harness, and we carry in no number from another paper. Arms are grouped by whether the update may move the vision–language backbone. The dotted lines mark the random-motion floor and the base policy. The hatched bar is the arm the simulator hands the target's coordinates at test time. Section 5 reads the figure.

(Pitis et al., 2020; Ameperosa et al., 2025). Point injection hands the action head a target it did not compute (Tsai et al., 2026; Yu et al., 2025), and attention alignment constrains where that head looks (Jia et al., 2026). Counterfactual guidance leaves the weights alone and edits the action at sampling time (Fang et al., 2026b). Each of them reports on a different base policy, with different demonstrations, against a different benchmark. The published record therefore cannot say which ingredient carries the repair.

We looked inside the policy first. It pairs a vision–language backbone with an *action expert*, a separate stack that emits the action chunks and reads the backbone through per-layer cross-attention. The keys and values that expert draws from each layer are its entire access to what the instruction means. We replace the keys and values from one band of backbone layers, leave every other layer alone, and ask whether the emitted action changes its target. One band moves it. Patching layers 18 to 23 flips the chosen object on 29 of the 32 pairs the base itself tells apart, and no other band flips more than one. We call those six layers the *read-out band*, the channel through which the action expert learns which object the instruction names. The repaired policy reads from the same band, on 76 of 78 pairs, and opens no other.

The band divides the repairs into two kinds. An update that freezes the backbone delivers into the band what the base delivered before; an update that reaches the backbone changes it. We registered both readings of that split before running any arm (Section 4), then varied one thing at a time. We fix one base policy, one demonstration budget, one training schedule and one evaluation, and run sixteen arms through them at n=200 trials per test condition, with three seeds on every arm that carries a headline claim. The ruler is a counterfactual that a position shortcut cannot pass: hold the pixels, the scene and the robot pose, change one word of the instruction, and record which object the end-effector reaches, against a random-motion floor of 12%.

The sixteen arms separate by where the update lands (Figure 1). Given only what the robot observes, every arm that leaves the backbone frozen stays inside 53%, whether it rewrites the action expert at rank 32 or at full rank, and whether it trains on the original demonstrations or on *decorrelated* ones, in which every object takes its turn as the target across randomized positions so that only the instruction predicts it. One frozen-backbone arm passes that ceiling, and it does so on coordinates

the simulator supplies at test time. Several of them also cut success on the trained tasks by more than half. Reaching the backbone is necessary and not sufficient: the same update, trained on demonstrations that keep each target in its accustomed position, reaches 44.6%. Decorrelating the demonstrations as well takes the rate to 97.7% [95.9, 98.7] across three seeds.

Contributions.

1. **The read-out band.** We locate the action expert's access to object identity in six consecutive backbone layers, on the checkpoint the repairs are applied to, with the layer indices, the trial counts and the flip criterion reported. The repair widens that band and opens no other (Section 4).
2. **A controlled comparison.** We run fifteen published and ablated repairs on one base policy, one demonstration set and one protocol, with three seeds on every arm that carries a headline claim, and show that the locus of the update decides the outcome (Section 5).
3. **Transfer off the checkpoint.** We show the surviving recipe holds on a second base, on a third-party counterfactual benchmark, on objects the demonstrations never made targets, on instructions never phrased that way, and on objects whose name and mesh never appear in training (Section 6). We release the policies, the demonstrations and their generator, and the measurement code.

2 Every repair arrives on its own base and its own ruler

Manipulation suites score motor competence. The standard suites place the named object where the demonstrations put it (Liu et al., 2023), so a policy that resolves the name and a policy that reaches a remembered position emit the same trajectory. The counterfactual re-releases of the last two years break that placement. Renaming or displacing the target drops open policies to near zero on object identity (Zhou et al., 2025; Fei et al., 2025), target selection sits at chance once grasp success is partialled out (Yu et al., 2026), and a thirty-policy sim-and-real evaluation puts the best open-vocabulary instruction-following score at 1.67% (Chen et al., 2026). We take this diagnosis as established.

Three families of repair, and no comparison between them. The families differ in where they act. The architectural family gives the policy a target it did not compute, routing an explicit 3D point or box into the action head (Tsai et al., 2026; Yu et al., 2025) or constraining the action expert's attention onto the referent (Jia et al., 2026). The data family removes the confound from the demonstrations, relabelling the instruction to match another object in the same scene (Glossop et al., 2025) or generating role-balanced counterfactual transitions (Pitis et al., 2020; Ameperosa et al., 2025). The inference family trains nothing and contrasts the instructed and uninstructed action at sampling time (Fang et al., 2026b; Zhang et al., 2026). A fourth proposal locates instruction following in the flatness of the optimum (Zhang & Bisk, 2026). Each reports on a different base, with different demonstrations, against a different benchmark. Section 5 runs all four on one base, and sorts them by whether they change what the read-out band carries.

Mechanistic evidence arrives beside the policies that get repaired. Activation-level work shows that language-conditioned behaviour in VLAs can be steered from the residual stream (Häon et al., 2025), and masking studies establish that the instruction is decorative at the level of outputs (Fang et al., 2026b; Lian et al., 2026). Researchers usually gather both on whichever model is small enough to instrument, then invoke them to explain a repair demonstrated on a different model. We measure the circuit on the checkpoint we repair, run the same protocol before and after, and report the layer indices, the trial counts and the flip criterion.

3 With no usable instruction the base still finishes half its benchmark

Base, demonstrations, evaluation. Every arm starts from `MolmoAct2-Think-LIBERO` (Fang et al., 2026a), an open policy already fine-tuned on LIBERO (Liu et al., 2023). Every trained arm

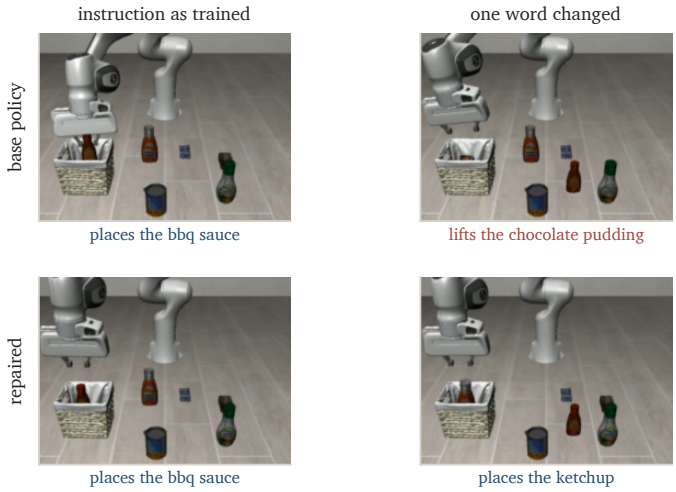


Figure 2: The instrument, on one LIBERO-Object slot. All four panels start from the same scene and the same robot pose, and one word of the instruction separates the columns, *bbq sauce* from *ketchup*. The line under each panel is that episode's own outcome. Frames come from the released clip, cropped and otherwise unaltered; Appendix B gives the rule that selected the slot before we rendered anything.

sees 900 demonstrations, takes 3000 updates at an effective batch of 8, and meets the ten object slots of LIBERO-PRO (Zhou et al., 2025), whose task files come from a third party and not from our generator. Arms differ in the demonstrations they see and in which parameters the update may move. Appendix A gives the setting table, including the learning rates the training code applies at run time.

Changing one word, with the scene held fixed. SWAP-OBEY holds the pixels, the scene and the robot pose, renames the target to another object that is physically present, and records which object the end-effector reaches. We call that renamed object the *decoy*. The decoy advances through every in-scene non-target from trial to trial, so no arm passes by orienting towards one familiar object. We score SWAP-OBEY on the 160 trials per seed whose decoy is not the scene's own original target, because in the other 40 the old habit and the new name point at the same place. Pooling all 200 would raise every arm by about twenty points.

Four control faces, and the floors we read them against. OCC blanks both cameras, and success must collapse. NONSENSE replaces the instruction with XXX. UNPERTURBED runs the original task untouched and prices what a repair costs. A do-nothing policy and a random-motion policy run the whole battery: on SWAP-OBEY the random-motion policy scores 12% at the 2 cm threshold and 20.5% at 5 cm, and the do-nothing policy scores zero at both. Every simulated rate below therefore reads against 12% and not against zero. A lift gate decides task success, so a scene whose target begins at the goal cannot score by standing still. We serve each policy under the input contract that favours it, the base with its native proprioceptive state and the repaired arms with a zeroed state, and a paired control confirms that this choice leaves their numbers where they are. We report every rate as a count out of its n, and every table carries a Wilson 95% interval per cell.

We read choice and competence apart. Alongside each rate we log the median distance from the end-effector to the named object and to the original target, which object the gripper first closes on, and which object leaves the table first, so a low task rate reads as a wrong choice or as a dropped grasp. Those distances separate by a factor of thirty on a repaired policy (Appendix B).

What the base reads. Row one of Table 1 gives the base policy's own numbers, which published comparisons on this axis normally leave blank. It completes the original tasks on 198 of 200 trials. Rename the target and it lands on the named object 57 times in 160, against a floor of 12%.

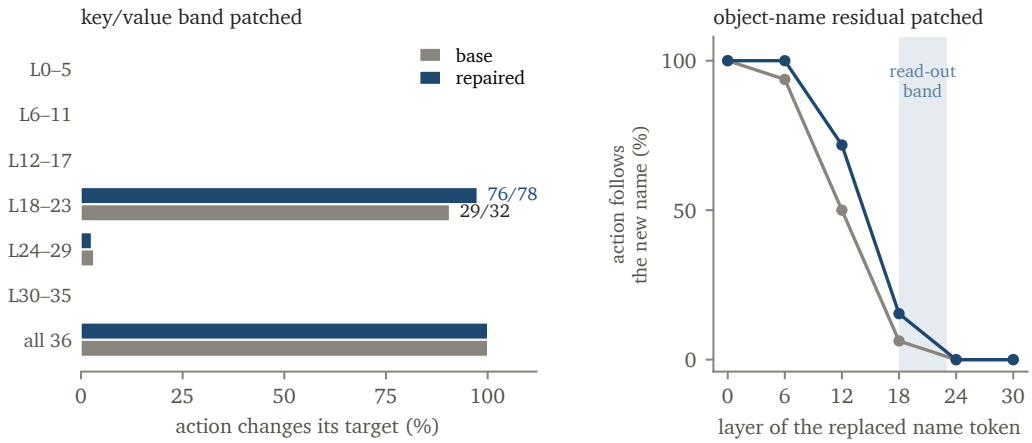


Figure 3: **Left:** we replace the keys and values the action expert draws from one band of backbone layers, leave every other layer alone, and ask whether the emitted action changes its target. Patching all 36 layers is the pipeline's positive control. **Right:** we replace the residual at the object-name token at increasing depth; the shaded span marks the read-out band from the left panel. Table 5 gives the flip rate and switch ratio for every band and depth.

It completes the renamed task 26 times in 200, with 20 of those in one slot, lifts nothing on 40 blanked-camera trials, and under `xxx` on the counterfactual scenes completes 1 of 200. Run the *stock* benchmark under `xxx` and it still completes 89 of 200. The 109-trial gap between 198 and 89 is what the instruction is worth to this policy on its own benchmark; the released checkpoint trained with the recipe of Section 5 completes 27 of 200 there, against 180 with the real instruction. Appendix C gives every arm on every face, and reads the two NONSENSE faces apart.

4 THE ACTION EXPERT READS OBJECT IDENTITY FROM SIX BACKBONE LAYERS

The backbone separates the candidates, and a frozen backbone keeps that separation intact. A swap-differential attention probe reads, for each of 100 paired scenes, whether the object-name tokens send more attention to object X's image region when the instruction names X than when it names the other object. Chance is 50%. The base separates the pair on 95 of 100 scenes, with the signal in layers L13–L17. Our repaired policy separates it on 100 of 100, strengthening that band and adding a second at L21–L26. The arm whose backbone we froze and whose action expert we trained instead reads 95 of 100, the base's own number, and tracks it within one to two percent scene by scene. Blanking both cameras drops the effect size by one to two orders of magnitude and removes the L13–L17 band, leaving a word-to-typical-position prior at 70–72% with no image behind it.

One band carries the choice. The action expert reads the backbone through per-layer cross-attention, so the keys and values it draws from each layer are its whole access to the instruction. For each scene we run two prompts, one naming X and one naming Y, matched to equal token length so the two forward passes align position by position. We then replace, in the X pass, the keys and values of one band of backbone layers. A trial flips when the resulting action block points at Y. We keep the 32 of 80 trials whose clean actions already distinguish the two objects, since a trial the base cannot resolve cannot show a flip, and we call those the *valid* trials. Patching all 36 layers reproduces the clean Y action bit for bit on every trial.

The result is a single band. Patching L18–23 flips the chosen object on 29 of 32 trials, 90.6% [75.8, 96.8], at a mean switch ratio of 0.89, where 0 leaves the action untouched and 1 replaces it with the alternative action in full. No other band flips more than one trial, and the switch ratio elsewhere stays under 0.07 (Figure 3, left). We call L18–23 the *read-out band*. Replacing the residual stream

Table 1: Fifteen repairs on one base policy, one demonstration budget (900 episodes), one schedule (3000 updates, effective batch 8) and one instrument. SWAP-OBEY counts trials in which the end-effector comes within 2 cm of the newly named object's origin, out of the 160 per seed whose renamed decoy is not the scene's own original target; the random-motion floor at that threshold is 12%, and a 5 cm threshold returns the same verdicts. TASK is completion of the renamed task and UNPERTURBED completion of the original task, both out of 200. Multiple entries are seeds 1000 / 2000 / 3000; pooled rates carry Wilson 95% intervals. OCC: no arm in this table lifted an object on any of its 40 blanked-camera trials. Appendix C repeats every arm on all five faces, including NONSENSE.

Repair	Demonstrations	SWAP-OBEY / 160	pooled %	TASK	UNPERTURBED
Vision–language backbone frozen					
Base policy, no update	none	57	35.6 [28.6, 43.3]	26	198
Counterfactual guidance w=1.5	none (inference)	68	42.5 [35.1, 50.2]	42	198
Counterfactual guidance w=2.0	none (inference)	80	50.0 [42.3, 57.7]	39	189
Counterfactual guidance w=2.5	none (inference)	85	53.1 [45.4, 60.7]	43	190
Action expert only, rank 32	confounded	61 / 63 / 63	39.0 [34.7, 43.4]	17 / 9 / 14	156 / 147 / 146
Action expert only, rank 32	position-randomized	64 / 57 / 51	35.8 [31.7, 40.2]	13 / 6 / 5	71 / 37 / 35
Action expert only, full rank	conf. / randomized	60 / 60	37.5 [30.4, 45.2]	8 / 13	154 / 76
Oracle point → action head	position-randomized	80	50.0 [42.3, 57.7]	18	102
same, post-hoc tuned	position-randomized	155	96.9 [92.9, 98.7]	83	142
Backbone adapted: rank 32 on the backbone and the action expert					
Sharpness-aware update, rank 32	confounded	77	48.1 [40.5, 55.8]	45	n/a
Same update, confounded data	confounded	76 / 70 / 68	44.6 [40.2, 49.1]	38 / 50 / 42	191 / 193 / 192
Counterfactual relabelling	fixed layout, relabelled	135 / 134 / 123	81.7 [78.0, 84.9]	114 / 95 / 98	131 / 127 / 120
Position randomization	position-randomized	**155 / 157 / 157**	**97.7 [95.9, 98.7]**	181 / 189 / 188	188 / 178 / 191
+ attention alignment, W=0.05	position-randomized	155	96.9 [92.9, 98.7]	168	175
+ attention alignment, W=0.2	position-randomized	154	96.2 [92.1, 98.3]	156	181

at the object-name token locates where identity stops living on the word itself: it decides the action at L0 and L6, decides half the trials at L12, and decides none by L18. That is the depth where the read-out band begins, and the depth of the attention signal above.

The repaired policy reads from the same band. The identical procedure on the repaired policy returns L18–23 and nothing else. That band flips 76 of 78 valid trials, 97.4% [91.1, 99.3], at a switch ratio of 0.96, and every other band stays at or below two. The count of valid trials reads the repair as well: the base's clean actions distinguish the two candidate objects on 32 of 80 pairs, and the repaired policy's on 78 of 80.

Takeaway. The knowledge sits in the backbone and the action expert already has a channel to it, so an update that freezes the backbone leaves that channel unchanged. Section 5 runs the preregistered test on sixteen arms.

5 Only a repair that changes the read-out band installs instruction following

Table 1 holds the comparison. Every row starts from the same checkpoint, sees 900 demonstrations, takes 3000 updates at an effective batch of 8, and meets the same instrument on the same ten slots, grouped by whether the update may move the backbone and so change what the read-out band carries. No arm lifted an object on any of its 40 blanked-camera trials.

A frozen backbone caps the repair at the base. Every arm in the upper block that works from the policy's own inputs reads within eighteen points of the base's 35.6%, and most read within four. Training the action expert on decorrelated demonstrations returns 35.8% [31.7, 40.2] over three seeds, which is the base's own rate. Releasing the rank, so the whole action expert moves, returns 37.5% [30.4, 45.2]. Steering the frozen policy at inference with counterfactual guidance climbs with the guidance weight and then flattens, at 42.5, 50.0 and 53.1% for w=1.5, 2.0 and 2.5. That 53.1%

[45.4, 60.7] is the highest rate any frozen-backbone arm reaches from the policy's own inputs, and the arm that reaches it does no training.

Two of these arms pay for the attempt. On decorrelated demonstrations the action-expert-only arms hold the named object no better than the base and drop the original tasks from 198 of 200 to 71, 37 and 35 across seeds; on the original confounded demonstrations they keep more of that competence, at 156, 147 and 146, and still fail to follow the instruction. Their end-effector settles a median 27, 57 and 54 mm from the named object, against 22 mm for the base and 3 mm for the recipe, so they fail where the base fails instead of collapsing into noise.

Supplied target coordinates raise the reach rate only. The architectural family's strongest form hands the action head the named object's true 3D coordinates at test time. Once the point pathway trains, that arm reaches the named object on 155 of 160 trials, and its competence stops there. It completes 83 of 200 renamed tasks where the recipe completes 181 to 189, holds 142 of 200 original tasks where the recipe holds 178 to 191, and with both cameras blanked drives the gripper to the supplied coordinates on 33 of 40 trials while lifting nothing. The coordinates move the end-effector to the right place and give the policy no use of the image. Under the preregistered point-encoder learning rate the pathway trains to a weight norm of 0.51 and the arm reads 50.0%, with the rest of the frozen-backbone block.

With the locus held fixed, the demonstrations decide the rate. The lower block holds the update at rank 32 over backbone and action expert together, and varies only the demonstrations. Trained on demonstrations that keep every target in its accustomed position, that update reaches 44.6% [40.2, 49.1]: the capacity to change the read-out band, spent on data that still pays for position, buys nine points. Relabelling the instruction across a fixed layout, so each in-scene object takes its turn as the named target, reaches 81.7% [78.0, 84.9]. This is a faithful port of counterfactual relabelling to this benchmark. Randomizing where the objects sit, so that the instruction becomes the only variable predicting the target, reaches 97.7% [95.9, 98.7]. Relabelling recovers 37.1 of the 53.1 points between the confounded arm and ours, and randomizing position recovers the remaining 16.0. The two data recipes share the base, the update, the 900 episodes, the oracle controller and the schedule, run at three seeds each, with per-seed intervals disjoint in every seed and a pooled Fisher $p{=}1.5\times10^{-17}$.

The two factors are jointly necessary. The 2×3 block inside Table 1 crosses two demonstration sets with three update loci. Exactly one cell rises above the base: decorrelated demonstrations with an update that reaches the backbone. Decorrelated demonstrations under a frozen backbone give 35.8% and 37.5%. Confounded demonstrations under a backbone-side update give 44.6%.

A flatter optimum does not substitute for the data. A recent proposal locates instruction following in the sharpness of the solution (Zhang & Bisk, 2026; Foret et al., 2021). Holding the confounded demonstrations and our recipe fixed and replacing only the update rule returns 48.1% [40.5, 55.8] against the same-seed Adam arm's 47.5%: a recovered fraction of 0.013, bootstrap $[-0.117, 0.127]$, paired exact McNemar $p{=}1$ over 200 matched trials. The optimizer did its job, since the sharpness-aware arm is flatter than its Adam counterpart on both fixed frame sets, and the arm that follows instructions is the sharpest of the three. At this rank, this budget and $\rho \in \{0.03, 0.075\}$, flatness and instruction following move independently. Attention supervision is equally inert once the backbone can move: the alignment objective on top of the winning recipe leaves the named-object rate at 96.9% and 96.2% against 97.7% and costs task completion.

Takeaway. The locus decides whether a repair works and the demonstrations decide how far it goes. Eight of the nine arms that leave the read-out band unchanged stay within eighteen points of the base, across three published families and two update ranks; the ninth needs the simulator to supply the target's coordinates.

6 THE SURVIVING RECIPE HOLDS OFF THE CHECKPOINT IT WAS TUNED ON

We took the recipe off the setting we built it on in five directions, listed in Table 2.

Table 2: What each transfer holds out, and what the base and the repaired policy read on it. Rows 1 and 6 are three seeds; rows 2–5 are single checkpoints on the stated n. Row 2 is our released checkpoint, which carries an additional auxiliary head (Section 6); the other rows are retrained arms.

Transfer	Held out	n	Base	Repaired
Second base policy ($\pi_{0.5}$)	architecture and weights	160×3	35.6%	**76.9%** [72.9, 80.4]
Third-party benchmark (LIBERO-CF)	scenes, wording, scoring	1300 ep.	380 instructed / 400 old	**575** instructed / **85** old
Objects never a target	3 object identities	60×3	n/a	**85.6%** [79.7, 89.9]
Unseen sentence forms	the phrasing	200	24/200	**179/200** (181 canonical)
Appearance-only description	the object's name	200	13/200	**114/200**
Unseen name *and* mesh (RoboCasa)	the object class	574	20% chance	**41.3%** [37.3, 45.4]

A second base. We moved the recipe onto $\pi_{0.5}$ (Physical Intelligence, 2025) unchanged: rank 32 on all linear layers of backbone and action expert, 5e-5, 3000 updates, the same 900 episodes. That policy's action expert attends jointly instead of through per-layer cross-attention. We fixed five acceptance criteria in writing before training, including a requirement that the base clear 180 of 200 original tasks under our service, and all five pass on all three seeds. The named-object rate moves from the base's 57 of 160 to 127, 123 and 119, pooled 76.9% [72.9, 80.4], with a paired exact McNemar $p \leq 3.6 \times 10^{-16}$ on matched trials. Original-task completion moves from 199 of 200 to 182, 184 and 183, and the substituted task reaches 72 of 200 where this base scores 20.

A benchmark we did not write. We ran the released LIBERO-CF suites (Fang et al., 2026b), whose scenes, instructions and scoring conditions all come from a third party, on the base and on our released checkpoint: five suites, 65 tasks, 1300 episodes per model, zero simulator faults. That checkpoint trains on the same position-randomized demonstrations under the same rank-32 update, and it additionally carries an auxiliary region head supervised during training by the named object's future segmentation mask. No arm in Table 1 carries that head, and nothing is supplied to the policy at test time beyond its cameras and the instruction. The two models invert. On matched tasks and matched initial states the base completes the instructed task 380 times and the task its demonstrations taught it 400 times; the repaired policy completes the instructed task 575 times and the demonstrated one 85. Over its four suites, instructed-task completion rises from 20.8% to 47.8% while completion of the demonstrated task falls from 41.5% to 9.1%. On its object suite its authors report 2.0, 6.0 and 11.4% for three other policies; the repaired policy completes 173 of 200.

Objects the demonstrations never targeted. We dropped every episode whose target was one of three objects, kept those objects in every scene as distractors, and retrained at three seeds. Asked to fetch them, the policy succeeds on 43, 52 and 59 of 60, pooled 85.6% [79.7, 89.9], against 96.1% for the same slots when the object was also demonstrated as a target, and it obeys them as the decoy on 46, 47 and 47 of 48. Appearing in a training scene as a distractor is therefore enough, at a cost of 10.5 points.

Instructions never phrased that way. Holding the scene, the initial state and the scoring fixed, we changed only the instruction string. Three unseen sentence forms that keep the object's name give 179 of 200 against 181 for the training phrasing, paired p=0.86. Describing the object by appearance alone, in words that occur in no training instruction such as "the red bottle with a white cap", the repaired policy completes 114 of 200 where the base completes 13, and six of the ten descriptions land.

Objects whose name and mesh never appear. The sharpest form of the question needs a wider object pool, so we built it in RoboCasa (Nasiriany et al., 2024). Of 64 screened graspable classes, 16 held-out classes share no word, no mesh and no look-alike group with any training class; five objects stand on the counter per scene and we draw the named one uniformly, so ignoring the name selects correctly 20% of the time. Trained on 48 classes and asked for a held-out one, the policy closes its gripper on the named object 237 times in 574 across three seeds, 41.3% with a Wilson lower bound of 37.3%, and blanking the cameras takes that to 4 of 120 with no successful grasp.

Takeaway. The behaviour survives a change of base policy, a change of benchmark author, and objects and phrasings the demonstrations never carried, and it reaches 41.3% against a 20% floor on classes whose names and meshes never appeared.

7 THE SAME RULER ON A PHYSICAL ARM: 96 OF 96

The recipe changes demonstrations, so it moves to hardware with no simulator in the loop. We teleoperated demonstrations on a low-cost SO-101 arm with a fixed overhead camera and a wrist camera, randomizing where each object sits, and applied the same rank-32 update to `MolmoAct2-SO100_101`. Four objects of different shape and colour stand on the table, and we run SWAP-OBEY exactly as Section 3 defines it: the layout stays where it is, the arm starts from the same pose, and each object is named once, so every layout is a complete four-way counterfactual. Twenty-four layouts, twelve on each of two cloth backgrounds, give 96 trials. Chance here is a ceiling, not a measured floor: within one layout a policy that ignores the word goes to the same object all four times and can be right at most once, so it scores at most 25%.

The repaired policy reaches the named object in every trial. Across all 24 layouts and all four words the arm goes to the object the instruction names, 96 of 96, [96.2, 100], and lifts it on 90 of 96, [87.0, 97.1]. All six misses reach the correct object and lose it at the grasp: two tissues tipped over on one background, and three bags and one box on the other. The same base policy without our update reaches the named object on 0 of 18 trials and lifts nothing (Fisher p=2.5×10^{-21}); its final waypoint barely leaves the home pose, with a pan span near 12° across objects and layouts, where the repaired policy sweeps about 90° to wherever the named object stands. This is the same measurement the simulated battery reports at 97.7% over three seeds, now on hardware.

The selection runs on the image. We blanked both camera streams and changed nothing else, neither the weights nor the layout nor the instruction, and the repaired policy lifts nothing on 24 of 24 trials, [0.0, 13.8]. Replacing the instruction with `xxx` on the same weights closed the gripper on some object 7 times in 10, spread across all four objects with no default, and on nothing 3 times. We report that spread instead of a success rate, since a nonsense instruction names no correct target.

These four objects differ in shape and in colour, which keeps the measurement on instruction following and leaves the separation of decorrelated from confounded demonstrations to Section 5, at 200 trials per face and three seeds per arm.

8 LIMITATIONS

The comparison of Table 1 varies the locus of the update on one base policy, and the second base carries the winning recipe with its own controls. Full-model fine-tuning is absent for a hardware reason: fully sharded training with offload drove host memory to 170 GiB on our machine, and without offload the fp32 parameters, gradients and two optimizer moments require 87 GB, so we make no claim about how it compares. The demonstrations randomize object position while holding the camera pose and the initial arm pose fixed, so the grounding they install is semantic, and we claim no viewpoint, clutter or lighting robustness. On hardware the unrepaired base is unadapted to our arm as well as to the task, so its 0 of 18 is a no-adaptation floor and not an isolation of grounding.

9 CONCLUSION

Instruction blindness in this policy is a property of where an update lands. The backbone separates the candidate objects, the action expert reads that separation through six consecutive layers, and an update that freezes those layers leaves them carrying what they carried before. Every repair that froze them, and worked from the robot's own observations, stopped at the base, across three published families and two update ranks. What moved the policy was an update that reached the backbone, trained on demonstrations in which only the instruction predicts the target: 36% to 97.7% on one base, 36% to 77% on a second.

Reproducibility statement

Section 3 and Appendix A give the base checkpoint, the 900-episode demonstration budget, the update schedule, the learning rates the training code applies, and the adapter placement for every arm in Table 1; Appendix B gives the scoring thresholds, the decoy rotation, the lift gate and the floors. The evaluation task files are third-party releases. Every arm reports its per-seed counts, so each pooled rate recomputes from the table. The supplementary material holds the evaluation harness and the serving contracts, and the collectors for the position-randomized and confounded demonstration sets. We release the repaired policies, the remaining baseline implementations, the patching code and the per-episode logs behind every count with the camera-ready version.

AI use statement

We used a large language model (Anthropic Claude) to draft and revise the text of this manuscript and to write part of the released experiment and evaluation code. Every number reported here comes from a logged run released with the paper, and the authors are accountable for all content.

References

Ezra Ameperosa, Jeremy A. Collins, Mrinal Jain, and Animesh Garg. RoCoDA: Counterfactual data augmentation for data-efficient robot learning from demonstrations. In *ICRA*, 2025. arXiv:2411.16959.

Tianxing Chen, Yue Chen, Zixuan Li, Junyuan Tang, Kailun Su, Haoran Lu, et al. Robodojo: A unified sim-and-real benchmark for comprehensive evaluation of generalist robot manipulation policies. *arXiv preprint arXiv:2607.04434*, 2026.

Pim de Haan, Dinesh Jayaraman, and Sergey Levine. Causal confusion in imitation learning. In *NeurIPS*, 2019.

Haoquan Fang, Jiafei Duan, Donovan Clay, et al. MolmoAct2: Action reasoning models for real-world deployment. *arXiv preprint arXiv:2605.02881*, 2026a.

Yu Fang, Yuchun Feng, Dong Jing, Jiaqi Liu, Yue Yang, Zhenyu Wei, Daniel Szafir, and Mingyu Ding. When vision overrides language: Evaluating and mitigating counterfactual failures in vlas. *arXiv preprint arXiv:2602.17659*, 2026b.

Senyu Fei, Siyin Wang, Junhao Shi, Zihao Dai, Jikun Cai, Pengfang Qian, Li Ji, Xinzhe He, Shiduo Zhang, Zhaoye Fei, Jinlan Fu, Jingjing Gong, and Xipeng Qiu. LIBERO-Plus: In-depth robustness analysis of vision-language-action models, 2025. arXiv:2510.13626.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*, 2021.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020. arXiv:2004.07780.

Catherine Glossop, William Chen, Arjun Bhorkar, Dhruv Shah, and Sergey Levine. CAST: Counterfactual labels improve instruction following in vision-language-action models, 2025. arXiv:2508.13446.

Bear Häon, Kaylene C. Stocking, Ian Chuang, and Claire Tomlin. Mechanistic interpretability for steering vision-language-action models. In *CoRL*, 2025. arXiv:2509.00328.

Xiaosong Jia, Bowen Yang, Zuhao Ge, et al. GuidedVLA: Specifying task-relevant factors via plug-and-play action attention specialization. *arXiv preprint arXiv:2605.12369*, 2026.

Shijie Lian, Bin Yu, Xiaopeng Lin, Laurence T. Yang, Zhaolong Shen, Changti Wu, Yuzhuo Miao, Cong Huang, and Kai Chen. LangForce: Bayesian decomposition of vision language action models via latent action queries. In *ICML*, 2026. arXiv:2601.15197.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. In *NeurIPS Datasets and Benchmarks*, 2023. arXiv:2306.03310.

Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, et al. RoboCasa: Large-scale simulation of everyday tasks for generalist robots. In *RSS*, 2024. arXiv:2406.02523.

Mohammad Pezeshki, Sékou-Oumar Kaba, Yoshua Bengio, Aaron Courville, Doina Precup, and Guillaume Lajoie. Gradient starvation: A learning proclivity in neural networks. In *NeurIPS*, 2021.

Physical Intelligence. $\pi_{0.5}$: a vision-language-action model with open-world generalization, 2025. arXiv:2504.16054.

Silviu Pitis, Elliot Creager, and Animesh Garg. Counterfactual data augmentation using locally factored dynamics. In *NeurIPS*, 2020. arXiv:2007.02863.

Shiang-Feng Tsai, Jin-Cheng Jhang, Yen-Ling Tai, Jia-Hong Lai, Shih-Yun Wong, Tung-Hsu Kang, and Yi-Ting Chen. Direct action-head injection of a grounded 3d point unlocks spatial and task generalization. *arXiv preprint arXiv:2606.27663*, 2026.

Bin Yu, Yao Zhang, Haishan Liu, Shijie Lian, Yuliang Wei, Xiaopeng Lin, Zhaolong Shen, Changti Wu, Ruina Hu, Bailing Wang, Cong Huang, and Kai Chen. RoboSemanticBench: Diagnosing semantic grounding in action prediction for vla models, 2026. arXiv:2606.02277.

Hang Yu, Juntu Zhao, Yufeng Liu, et al. Point what you mean: Visually grounded instruction policy. *arXiv preprint arXiv:2512.18933*, 2025.

Haochen Zhang and Yonatan Bisk. Flatness preserves instruction following in vision-language-action models, 2026. arXiv:2606.23641.

Yan Zhang, Yinan Wu, Haoran Duan, and Jungong Han. CofactVLA: Deconfounding vision-language-action models via counterfactual intervention, 2026. arXiv:2608.04396.

Xueyang Zhou, Yangming Xu, Guiyao Tie, Yongchao Chen, Guowen Zhang, Duanfeng Chu, Pan Zhou, and Lichao Sun. LIBERO-PRO: Towards robust and fair evaluation of vision-language-action models beyond memorization, 2025. arXiv:2510.03827.

A SETTING TABLE

Table 3 collects every setting shared by the arms of Table 1, including the learning rates the training code applies. The configuration field named `optimizer_lr` is not a learning rate for any parameter group in this code path: with low-rank adaptation enabled, the backbone, vision tower and connector groups are set to 5e-5 in the source, the action expert takes its own field, and `optimizer_lr` sets only the peak the cosine schedule decays from, so the floor is a tenth of it. Measured from the training log, warmup ends at update 200 (4.9e-5 at update 400) and the schedule reaches 5.2e-6 at update 3000.

Table 3: Settings shared by every trained arm in Table 1.

Base checkpoint	`MolmoAct2-Think-LIBERO`; action expert reads the backbone by per-layer cross-attention
Demonstrations	900 episodes, scripted oracle, per arm; 8-dimensional proprioceptive state
Adapter	rank 32, α 16, on every linear layer of the covered modules
Trainable parameters	backbone + action expert 106.2 M; action expert only 31.3 M
Learning rate	every group 5e-5, warmup 200 updates, cosine to 5e-6, weight decay 0
Schedule	3000 updates, 4-way data parallel $\times$ batch 2 = effective batch 8, gradient checkpointing
Seeds	1000, 2000, 3000
Evaluation	LIBERO-PRO `libero_object_task`, 10 slots, initial states 0–19, n=200 per face
Selection threshold	2 cm from the object origin; 5 cm reported as a check
Service contract	repaired arms zeroed state, base its native state; normalization from the checkpoint's own processor

The three demonstration sets. All three come from the same scripted oracle at the same episode count, and differ only in where the target sits and which object is named. *Position-randomized*: every object appears as the target and as a distractor, and target positions are randomized across the workspace, so neither position nor scene identity predicts the target and only the name does. *Confounded*: the same oracle collects the same 900 episodes with each scene's target in its accustomed position under a ± 2 cm jitter, and each scene contributes only its own original target, so the instruction is fully determined by the scene. Measured on that set, a nearest-centroid classifier recovers object identity from target position at 0.224 against a 0.100 chance level; the same collector without the jitter reaches 0.717, and we report both because the jitter makes the confounded arm a harder baseline, not an easier one. The load-bearing confound is unaffected either way: within a scene the confounded set carries one instruction and one target, so the in-scene instruction entropy is zero. *Counterfactual relabelling*: the original LIBERO layout with the same jitter and no position randomization, with each of the six in-scene objects taking its turn as the named target for 15 episodes per (scene, target) pair.

The full-rank action-expert arm. Its learning rate was selected before any test-set evaluation, from $\{1.5\text{e-}5, 5\text{e-}5, 1.5\text{e-}4\}$ at seed 1000, on validation initial states 25–34 that do not overlap the test states 0–19, n=100 per face, by task success with the named-object rate as the tie-break. Validation reads 16, 5 and 1 of 100 task successes and 34, 24 and 26 of 80 named-object successes, so 1.5e-5 was selected; it is the low end of the sweep, and the trend over the three is that the closer the rate is to zero the closer the arm is to the base. Training loss converges on all three (0.487 at update 200 to 0.018 at update 3000 for 5e-5).

B MEASUREMENT DISCIPLINE

The faces. UNPERTURBED runs the original task file unchanged. TASK and SWAP-OBEY run the counterfactual task file, in which one word of the instruction names a different in-scene object while the scene geometry and the robot's initial pose are bit-identical to the original. OCC runs the counterfactual file with both camera streams set to zero. NONSENSE replaces the instruction with xxx. Each face is 200 trials, 20 per slot, except OCC at 40.

Decoy rotation and stratification. The renamed decoy advances through the scene's non-target objects from trial to trial, so no arm can pass by orienting towards a single familiar object. In 40 of the 200 trials the decoy the rotation selects is the scene's own original target; those trials cannot separate the old habit from the new name and are excluded from every SWAP-OBEY rate in this paper, leaving n=160. For reference, the base scores 37 of those 40 and the repaired arms score 39–40 of 40, so pooling all 200 raises every arm by roughly twenty points.

Scoring and gates. A trial counts as a named-object success when the end-effector comes within 2 cm of the newly named object's origin. Alongside it we log the median distance to the named object and to the original target, which object the gripper first closes on, which object leaves the table first, and whether the failure reached the right object. Task success is lift-gated, so a scene whose target begins at the goal cannot score by standing still. Two mock policies run the whole battery on the identical geometry: a do-nothing policy scores 0 of 200 at both thresholds, and a random-motion policy scores 24 of 200 at 2 cm and 41 of 200 at 5 cm and never lifts an object, which is the 12% floor used throughout.

Distances. On the repaired arms the end-effector settles a median 2–3 mm from the newly named object and 93 mm from the object the demonstrations trained it to fetch, and the original target is fetched on 0 of 200 trials at every seed. On the base the same medians are 22 mm and 73 mm. The threshold is therefore not doing the work: a 5 cm criterion returns the same verdicts.

How the figure's scene was chosen. The selection rule for Figure 2 was fixed before any frame was rendered. Of the ten LIBERO-Object slots we keep those in which, over the n=200 evaluation, the base completes the changed-word task on 0 of 20 start states, the repaired policy on 20 of 20, and both complete the original task on 20 of 20. Three slots qualify, and the figure shows the first in the benchmark's order. All four panels use that slot's first test start state, and the original and

changed-word task files place every object identically on it. The frames are cropped and otherwise unaltered.

C EVERY ARM ON EVERY FACE

Table 4: Every arm on all five faces, answering per cell with its own count. n=200 per face except OCC at n=40; multiple entries are seeds 1000 / 2000 / 3000. SWAP-OBEY is the count of the 160 informative trials per seed in which the end-effector comes within 2 cm of the newly named object's origin; the other four faces are counts of task completion under the lift gate. NONSENSE runs the counterfactual scene with the instruction replaced by `xxx`, where the scorer credits the substituted object, so it measures how often an arm reaches that object without being told to (Section 3).

Arm	Demonstrations	SWAP-OBEY/160	TASK	UNPERTURBED	NONSENSE	OCC
Base policy, no update	none	57	26	198	1	0
Vision–language backbone frozen						
Counterfactual guidance w=1.5	none (inference)	68	42	198	0	0
Counterfactual guidance w=2.0	none (inference)	80	39	189	0	0
Counterfactual guidance w=2.5	none (inference)	85	43	190	0	0
Action expert only, rank 32	confounded	61 / 63 / 63	17 / 9 / 14	156 / 147 / 146	0 / 0 / 0	0 / 0 / 0
Action expert only, rank 32	position-randomized	64 / 57 / 51	13 / 6 / 5	71 / 37 / 35	0 / 0 / 0	0 / 0 / 0
Action expert only, full rank	confounded	60	8	154	1	0
Action expert only, full rank	position-randomized	60	13	76	8	0
Oracle point, preregistered	position-randomized	80	18	102	3	0
same, post-hoc tuned	position-randomized	155	83	142	16	0
Backbone adapted: rank 32 on the backbone and the action expert						
Sharpness-aware update, rank 32	confounded	77	45	n/a	n/a	n/a
Same update, confounded data	confounded	76 / 70 / 68	38 / 50 / 42	191 / 193 / 192	4 / 8 / 5	0 / 0 / 0
Counterfactual relabelling	fixed layout, relabelled	135 / 134 / 123	114 / 95 / 98	131 / 127 / 120	1 / 0 / 0	0 / 0 / 0
Position randomization	position-randomized	**155 / 157 / 157**	181 / 189 / 188	188 / 178 / 191	28 / 25 / 18	0 / 0 / 0
+ attention alignment, W=0.05	position-randomized	155	168	175	4	0
+ attention alignment, W=0.2	position-randomized	154	156	181	0	0

D ACTIVATION PATCHING

Both interventions run on CPU in fp32, with each model loaded exactly as its evaluation server loads it, over the 50 scenes used for the attention probe. For each scene we form two prompts, one naming the task's target X and one naming the first in-scene object Y whose name occupies the same number of tokens, so that the two forward passes align position by position; the one three-token object name has no partner and its scenes are skipped, leaving 40 pairable scenes run in both directions for 80 trials. The read-out is the direction of the net xy displacement of one ten-step action block after denormalization, compared by cosine against the direction from the end-effector to each object, with one shared flow-matching noise draw per scene. A trial is *valid* when the clean X prompt yields an action pointing at X and the clean Y prompt one pointing at Y; the base satisfies this on 32 of 80 and the repaired policy on 78 of 80. A trial *flips* when the patched action's cosine to Y exceeds its cosine to X; we also report the continuous switch ratio $s = (a_{\text{patched}} - a_X) \cdot (a_Y - a_X)/|a_Y - a_X|^2$, which is 0 for no movement and 1 for the full Y action. Patching all 36 layers reproduces the clean Y action exactly on every trial.

Table 5 gives every band and depth. The intervention on the residual stream replaces, at the output of layer L, the activations at the object-name token positions, and lets every later layer recompute.

The attention probe. For each of 100 ordered scene pairs we take the attention from the object-name tokens to the agent-view image patches, averaged over heads and name tokens, assign each patch to its nearest object, and ask whether naming X raises the share falling on X's region relative to naming the other object. Chance is 50%. Base 95/100 with effects +0.045 and +0.057; repaired 100/100 with +0.078 and +0.101; the frozen-backbone arm 95/100 with the base's effects to within one to two percent per scene, although it runs the repaired models' 470-token prompt format and the

Table 5: Flip rate and switch ratio for every band and depth. Valid trials: base 32, repaired 78.

	Base		Repaired	
Intervention	flips	s	flips	s
Key/value band L0–5	0/32	0.000	0/78	0.000
Key/value band L6–11	0/32	0.003	0/78	0.000
Key/value band L12–17	0/32	0.033	0/78	0.001
Key/value band L18–23	**29/32**	**0.894**	**76/78**	**0.956**
Key/value band L24–29	1/32	0.068	2/78	0.041
Key/value band L30–35	0/32	0.000	0/78	0.000
All 36 layers (control)	32/32	1.000	78/78	1.000
Name-token residual, L0	32/32	0.998	78/78	1.000
Name-token residual, L6	30/32	0.900	78/78	0.999
Name-token residual, L12	16/32	0.472	56/78	0.786
Name-token residual, L18	2/32	0.147	12/78	0.136
Name-token residual, L24	0/32	0.000	0/78	0.000
Name-token residual, L30	0/32	0.000	0/78	0.000

base runs a 488-token one, which shows the prompt format is not what moves the signal. With both cameras zeroed all models read 70–72/100 at effects one to two orders of magnitude smaller and the L13–L17 band disappears.

E THIRD-PARTY BENCHMARK, PER SUITE

Table 6 gives the LIBERO-CF suites separately, scored by that benchmark's own conditions on 20 episodes per task. *Instructed* counts episodes in which the policy touches or completes the object the instruction names; *demonstrated* counts the object its training demonstrations taught it to fetch. The focused suite has no demonstrated-object condition and is excluded from the four-suite average, following the benchmark's own convention.

Table 6: LIBERO-CF, 1300 episodes per model, zero simulator faults. Percentages, Wilson 95% in the text.

		Base		Repaired	
Suite	n	instructed	demonstrated	instructed	demonstrated
spatial	300	52.7 / 27.7	45.3 / 41.7	72.3 / 27.0	31.0 / 8.3
object	200	25.0 / 15.5	50.5 / 46.5	93.5 / 86.5	2.0 / 1.0
long	200	55.0 / 18.0	61.5 / 51.0	74.5 / 33.0	63.0 / 23.5
ood	300	72.0 / 22.0	28.7 / 26.7	92.3 / 44.7	10.3 / 3.7
four-suite average		51.2 / 20.8	46.5 / 41.5	83.2 / 47.8	26.6 / 9.1
spatial-focused	300	92.0 / 54.7	n/a	94.0 / 40.3	n/a

Two entries deserve their own sentence. On the spatial suite the repaired policy completes 81 instructed episodes against the base's 83 while its demonstrated-task completions fall from 125 to 25 and its instruction-following touch rises from 52.7% to 72.3%: it is choosing correctly and losing at the placement. On the out-of-distribution suite, whose objects are absent from both models' training, the repaired policy completes 134 of 300 against the base's 66; the objects in those scenes are also the ones the base touches most often, so this number is consistent with instruction-following and with the release of an old habit alike, and we do not read open-vocabulary grounding from it. The RoboCasa experiment of Section 6 is the controlled form of that question.

F UNSEEN NAMES AND MESHES IN ROBOCASA

Of 112 graspable RoboCasa classes with at least three meshes, 97 survive removal of large containers and duplicate instruction words, 64 pass an oracle screen of at least two successes in four attempts

and a visual check at the rendered resolution. Sixteen of the 64 share no instruction word, no mesh and no look-alike group with any other class, and those sixteen are the held-out set; the remaining 48 and a five-class subset are the two training vocabularies. Each scene places five objects on the counter with at least 10 cm of separation and re-draws if any object settles into the sink, moves more than 1.5 cm at settling, or projects to fewer than eight visible pixels. Selection counts the object nearest the gripper when the closing command first persists for three control steps, within 5 cm horizontally and 10 cm vertically; on the oracle demonstrations that criterion identifies the target on at least 99.5% of episodes in every vocabulary. The named object is drawn uniformly from the five, so any policy that does not read the name scores at most 20%. Held-out selection is 237 of 574 at 48 classes and 283 of 575 at five, in-vocabulary selection 40.5% and 70.8%, and with both cameras zeroed 4 of 120 and 1 of 120 with no successful grasp in any of the six models.

G COMPUTE

Every training run in this paper is 3000 updates at an effective batch of 8 on four rented RTX 3090s. One arm of the recipe takes 1 h 27 min. The sharpness-aware arm takes 6 h 45 min, since it computes two forward and backward passes per update, and the second base takes 1 h 57 min. The comparison as a whole is 38 such runs, and the evaluation behind it is more than 30,000 simulated episodes, since every arm meets the full battery at n=200 per face and the multi-entry arms run it three times. Evaluation runs on a second machine, so training and serving never contend for the same card.

The work spans about three months on rented commodity hardware and cost on the order of US$1,000 in GPU time, counting the arms that failed, the sweeps that selected a learning rate, and the runs we discarded. We give the order of magnitude and not a precise figure: we rented and destroyed machines repeatedly over that period and did not meter them per experiment. A reader who wants one arm instead of the whole comparison needs the 1 h 27 min above, which is the unit this paper is built from.

The patching and probe experiments in Appendix D run on CPU and recompute from the released logs with no GPU.