WHEN IMAGES READ THE INSTRUCTION: BIDIRECTIONAL ATTENTION IN VISION-LANGUAGE-ACTION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Embodied reasoning and manipulation require rich interaction between visual, linguistic, and proprioceptive modalities, yet the dominant VLA backbone is a causal decoder that encodes the image before the instruction or robot state enters the context. Bidirectional attention is a natural remedy, as all tokens can freely attend to each other. Popular VLAs already adopt bidirectional observation encoding, but its effect has not been tested in a controlled comparison. In this work, we systematically study how bidirectional attention shapes representations for embodied tasks. We build matched paradigms from one causal Qwen3-VL-2B that vary in their embodied pretraining and attention mask: causal, flipped to bidirectional at policy time, or bidirectional and learned during pretraining. We show that only a learned attention mask improves manipulation over causal pretraining on the same data when using a final layer acton head. Activation patching, closed-loop instruction transplants, attention-path knockouts, and object-level attention measurements trace this difference. The attention mask decides where the instruction is represented: every bidirectional mask moves the instruction into the image tokens, and the policy reads its task from there. At what stage the mask is learned decides whether the instruction's attention stays on the task objects. A learned mask preserves the object selectivity that embodied pretraining creates, while a flipped mask does not. Our results suggest that bidirectional attention in VLA backbones should be aligned during pretraining rather than added at policy time.

1 INTRODUCTION

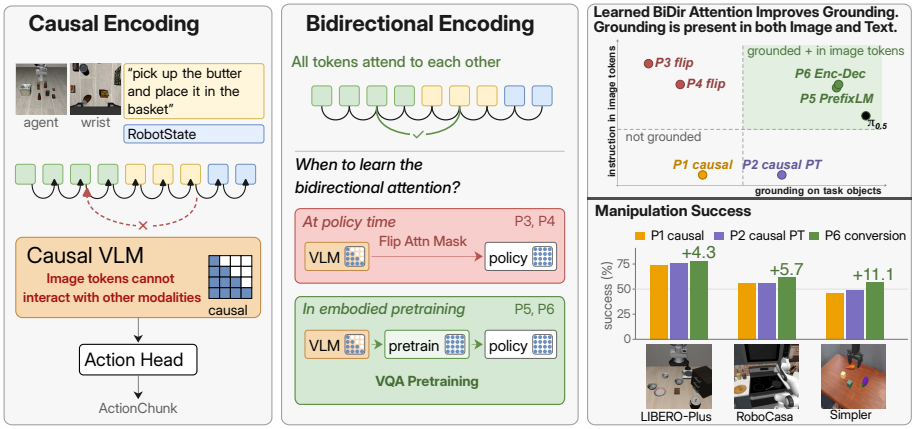


Figure 1: **With a final-layer action head, bidirectional attention helps manipulation when learned during embodied pretraining, whereas imposing it at policy time does not.** Left: a causal VLM encodes image tokens before the instruction and state. Middle: the mask can be imposed at policy time (P3, P4) or learned during pretraining (P5 PrefixLM, P6 conversion). Top right: schematic of Figure 4. Bottom right: success on LIBERO-Plus, RoboCasa365 pick-and-place, and SimplerEnv distractors.

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.

Embodied reasoning and manipulation require grounding the instruction in the observation before any action is produced: to pick up the left bowl, an agent must localise the referent and tie the instruction to the scene. Yet most vision-language-action models(VLAs) (Kim et al., 2024; Bjorck et al., 2025) are built on causal decoders inherited from text generation, which encode the image tokens before the instruction or the robot state enters the context (Figure 1, left). Whatever the instruction implies for the scene must therefore be recovered later in the sequence, never by the visual representation itself. Bidirectional attention is a natural remedy, since it lets image tokens condition on the instruction, and several prominent VLAs already use it: π_0 and $\pi_{0.5}$ (Black et al., 2025; Intelligence et al., 2025) build on PaliGemma's bidirectional prefix, UP-VLA (Zhang et al., 2025d) makes image tokens mutually visible, and F1 (Lv et al., 2025) uses bidirectional intra-expert attention. But does the bidirectional attention mask itself help, or is something else doing the work?

No existing comparison isolates the attention mask from the rest of the design. Encoder-decoder VLAs such as FLOWER (Reuss et al., 2025), ThinkProprio (Wang et al., 2026a), and X-VLA (Zheng et al., 2026) start from natively bidirectional backbones, so changing the mask means changing the model. Other methods impose bidirectional attention at policy time together with a new objective or architecture (Ding et al., 2026; Chen et al., 2026), and the π family inherits PaliGemma's mask together with its corpus and weights. In language modelling, PrefixLM (Raffel et al., 2019; Tay et al., 2022) pretrains under the new mask and conversion (Zhang et al., 2025a) adapts to it in a dedicated stage. Enabling bidirectional attention without either degrades most causal models (BehnamGhader et al., 2024). Whether the mask helps an embodied policy, and whether it must be learned first, therefore remains open.

To answer this question, we conduct a systematic study of how bidirectional attention masks change pretrained causal VLM representations and downstream policy learning. We build six matched paradigms from one causal VLM, Qwen3-VL-2B (Bai et al., 2025), by varying two factors (Figure 2): the embodied pretraining of the backbone (none, or 6.5M samples under a causal mask, a PrefixLM mask, or encoder-decoder conversion (Zhang et al., 2025a)) and the policy's attention mask (causal, bidirectional imposed at policy time, or bidirectional learned in pretraining). Every paradigm trains the same policy with the same data, action head, state tokens, schedule, and executed trainable parameters. We evaluate on LIBERO-Plus (Fei et al., 2026), RoboCasa365 (Nasiriany et al., 2026), and, for policies trained on real images, SimplerEnv (Li et al., 2024b), and locate the instruction inside each policy with activation patching (Zhang & Nanda, 2024), a closed-loop transplant, attention-path knockouts (Geva et al., 2023), and the attention on segmented task objects.

The two factors play distinct roles. Embodied pretraining decides *whether* instruction attention is selective for the task objects. The attention mask decides *where* the instruction's effect is represented: under a bidirectional mask, the image tokens carry most of it. Learning the mask during embodied pretraining improves manipulation, with the gains concentrated on tasks where the instruction must single out an object. Flipping the same mask at policy time without retraining under it makes the policy worse. The difference traces to two representational properties. Every bidirectional mask moves the instruction into the image tokens, and the policy reads its task from there: transplanting only the image tokens computed under another instruction makes the robot perform that other task in most episodes. A learned mask, however, keeps the instruction's attention object-selective, whereas a flipped mask destroys it. Only when both properties hold together does bidirectional attention help.

We contribute (i) a six-paradigm study of how embodied pretraining and the VLM attention mask interact at matched data, head, and executed parameters, (ii) converging evidence for the account above from patching, a closed-loop transplant, knockouts and object-level attention, and (iii) a recipe: learn the attention mask during embodied pretraining. The gain is established for one 2B VLM family with a final-layer head (Section 5).

2 Related Work

What VLAs inherit from VLMs. VLM4VLA (Zhang et al., 2026a) places many VLMs behind one action head and finds that general VLM skill predicts control poorly. Related studies find that action training erodes visual priors (Zhang et al., 2025c; Kachaev et al., 2025) and that backbone, action formulation, initialisation and data all shape control (Li et al., 2024a; Wu et al., 2026; Lin et al., 2026). These studies compare different pretrained models, not different masks on the same

model. Our causal-pretraining paradigm reproduces their headline, embodied data improving reasoning far more than control, and our probes are consistent with one reason for it: the pretrained causal model attends to the task objects, but its image tokens cannot represent the instruction.

The causal blind spot and its fixes. In a causal VLM the image tokens precede the text and cannot attend to the instruction or the state. Readout and latent-query tokens let the head read a task-conditioned summary after the prompt (Ding et al., 2026; Xiong et al., 2026; Wang et al., 2026b; Ma et al., 2026; Zhang et al., 2026b). Other approaches gate visual tokens before the VLM (Wang et al., 2026a) or learn latent actions with cross-view reconstruction (Lee et al., 2026) and add geometric supervision (Han et al., 2026). Structured masks are an established alternative (Black et al., 2025; Intelligence et al., 2025; Zhang et al., 2025d; Lv et al., 2025), but each changes the attention mask together with the backbone, the corpus, or both. To the best of our knowledge, we provide the first controlled test of when and why bidirectionality helps.

Attention mask ablations and causal-to-bidirectional adaptation. Several VLA studies have varied the attention mask, but each comparison also changes the model or the data: bidirectional attention over image and action tokens outperforms a causal mask on CALVIN (Chen et al., 2026), a full-attention prefix transfers better than autoregressive masking in PaliGemma (Beyer et al., 2024), and StreamPI (Liu et al., 2026) gains up to 2.8 points on LIBERO from bidirectional image-text attention. None of these results isolates the mask from other design choices. In language modelling, PrefixLM objectives (Raffel et al., 2019; Zhang et al., 2025b; Tay et al., 2022) and encoder-decoder adaptation (Zhang et al., 2025a) train under the new mask before downstream use, and LLM2Vec (BehnamGhader et al., 2024) shows that simply enabling bidirectional attention degrades most causal models until a short adaptation stage is added. Our flip and learned-mask paradigms are the embodied analogue of this contrast. Our probes show that the flip preserves the image-carried route but loses object-selective instruction attention, the property that makes the route useful.

Locating information in multimodal models. Attention knockouts (Geva et al., 2023) and activation patching (Meng et al., 2022; Zhang & Nanda, 2024) locate where a model stores or routes information, typically for next-token predictions of language models. We apply both to VLA policies and measure their effect on the predicted actions and on closed-loop success, together with an object-level attention measure on simulator segmentations.

3 Study Design

To study the effect of attention masks on VLAs, we separate the attention mask from the stage at which it is learned with six paradigms from one causal VLM (Sections 3.1–3.3) and locate the instruction inside each trained policy with test-time probes (Section 3.4).

3.1 Six paradigms from one causal VLM

Every paradigm starts from the same causal Qwen3-VL-2B (Bai et al., 2025) and is defined by two choices (Figure 2a): (i) the *embodied pretraining* of the backbone before policy learning: none, or one epoch on the same 6.5M samples with causal next-token training, PrefixLM training, or encoder-decoder conversion, and (ii) the *attention mask* of the policy backbone over the observation: causal or bidirectional. We chose Qwen3-VL-2B because it is widely used as a VLA backbone, uses only softmax attention, and is available in several sizes.

- **P1 (causal base).** No pretraining, causal mask: the standard VLA recipe.
- **P2 (causal pretraining).** Causal pretraining, causal mask.
- **P3 (flip).** No pretraining. The causal weights are deep-copied into a bidirectional encoder at policy time.
- **P4 (pretraining + flip).** P2's causally pretrained weights, flipped to bidirectional at policy time.
- **P5 (PrefixLM).** PrefixLM pretraining in a single stack, bidirectional prefix at policy time.
- **P6 (conversion).** Encoder-decoder pretraining. The decoder is discarded and the bidirectional encoder is the policy backbone.

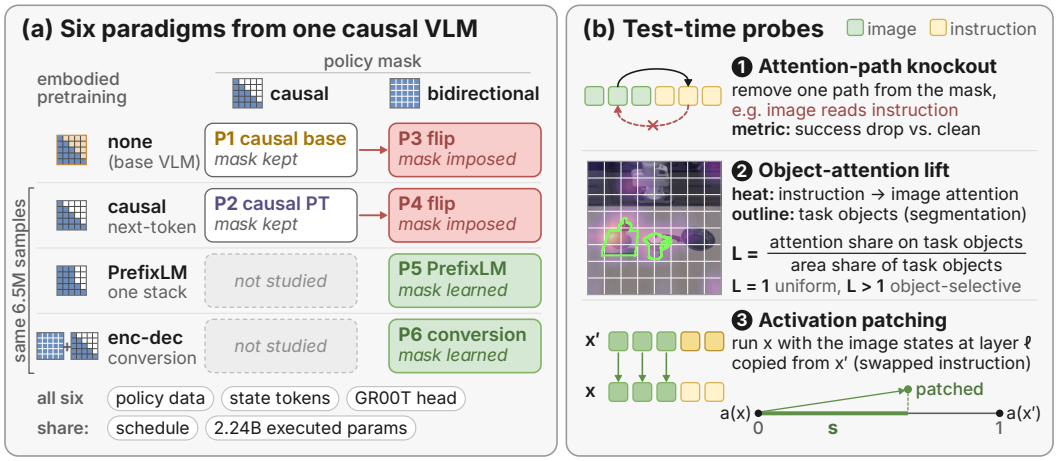


Figure 2: **Study design.** **(a)** Each paradigm is one cell of a grid over the embodied pretraining the backbone receives (rows) and the attention mask of the policy backbone (columns). The icons draw each mask. In the red cells the bidirectional attention mask is *imposed* at policy time on causal weights (flip). In the green cells it is *learned* during embodied pretraining. All six share the policy data and schedule, the state tokens, the GR00T-style action head, and the executed trainable parameters. The four pretrained paradigms share one 6.5M-sample corpus. **(b)** Three test-time probes: removing one attention path and measuring the success lost. The object-attention lift L of instruction attention on the segmented task objects (heat: P6 on a LIBERO-Plus frame), and activation patching, which copies the image-token states of a run with a swapped instruction and measures the share s of the action change they transmit.

We call P3 and P4 the flips and P5 and P6 the learned attention masks. Each pair isolates one factor at equal data. P2−P1 measures the embodied data under a causal mask. P3−P1 and P4−P2 measure a flip at equal weights, without and after pretraining. P5−P4 and P6−P4 compare a learned with an imposed attention mask, and P6−P5 compares the encoder-decoder with a single stack under the same learned mask.

3.2 Pretraining variants

All three variants train one epoch on the same corpus (the Embodied-Finetuning Mixture: 6.5M embodied-reasoning samples from 38 public datasets with next-token prediction on the answer tokens. More details are presented in Appendix C). *Causal pretraining* continues Qwen3-VL-2B with its original mask. *PrefixLM* (Raffel et al., 2019; Tay et al., 2022) keeps the single stack and changes only visibility: prompt and observation tokens attend to one another, and answer tokens attend to the prompt and causally to earlier answer tokens. *Conversion* follows T5Gemma (Zhang et al., 2025a;b): all 28 text layers are deep-copied with the causal flag removed to form a bidirectional encoder, and the originals remain a causal decoder that reads the encoder's final layer through the merged self- and cross-attention of T5Gemma 2. The variants therefore differ in the attention mask and, for conversion, in a second text stack during pretraining: PrefixLM adds no parameters, and the encoder-decoder has 3.5B as both stacks share one vision tower and one embedding table. For policy learning the decoder is discarded and the action head reads the encoder's final hidden states.

3.3 Matched policy learning

Every paradigm trains the same policy: a GR00T-style (Bjorck et al., 2025) flow-matching DiT head that here cross-attends to the backbone's final hidden states, with identical head configuration, augmentation, data order, schedule, and seeds within each benchmark (Section 4.1). Robot state enters only through the backbone, as discretised vocabulary tokens spliced after the instruction (Intelligence et al., 2025). Under a causal mask the image tokens therefore never see the instruction or the state, whereas under a bidirectional attention mask every observation token does.

3.4 MECHANISM PROBES

We investigate how bidirectional attention shapes the VLMs representations with four probes. Activation patching finds which tokens carry the what information, the closed-loop transplant and attention knockouts test what information the policy uses, and the object-attention lift measures what the VLM attends to.

Attention-path knockouts. To measure how much a policy depends on one information route, we use attention-path knockouts (Geva et al., 2023). Given a query group $\mathcal{Q}$ and a key group $\mathcal{K}$, we set the pre-softmax mask entries to $-\infty$ in every layer ℓ and head h,

$$M^{\ell,h}_{q,k} = -\infty \quad \forall\, q \in \mathcal{Q},\ k \in \mathcal{K}, \tag{1}$$

so that softmax renormalises over the remaining keys. The *reverse* knockout stops the image tokens from reading the instruction ($\mathcal{Q}$=image, $\mathcal{K}$=instruction), a route that exists only under a bidirectional mask. The *forward* knockout stops the instruction from reading the image, a route open under both masks. We roll out the policy on the same episodes with and without the knockout, and the *knockout drop*, clean minus knockout success, measures its dependence on the route.

Activation patching. Knockouts push the policy off its training distribution. Activation patching locates the instruction's effect without deleting anything (Meng et al., 2022; Zhang & Nanda, 2024). For a single frame we form a base input x and a source input x' that differs in one component (another task's instruction, or the state of a later frame). Let $\mathbf{a}(x)$ be the predicted action chunk under fixed flow-matching noise. We run x but replace the hidden states of one token group G at the output of layer ℓ with those from x', yielding $\mathbf{a}_{G,\ell}$, and report the fraction of the source's effect this transmits,

$$s_{G,\ell} \;=\; \frac{\langle \mathbf{a}_{G,\ell} - \mathbf{a}(x),\ \mathbf{a}(x') - \mathbf{a}(x) \rangle}{\|\mathbf{a}(x') - \mathbf{a}(x)\|_2^2}, \tag{2}$$

averaged over 120 to 160 frames. s=1 means the group alone reproduces the swap's effect and s=0 that it transmits none. We patch at layer 26, one layer before the head's input (layer 27). Under a causal mask s_{image}=0 exactly, a built-in sanity check. As a control, we also patch causal policies run under a bidirectional mask at probe time only (*probe-time flip*).

Closed-loop transplant. Patching shows what the image tokens carry, not whether the policy acts on it. On LIBERO-Goal, whose tasks share one scene, we therefore run the policy with its own instruction A but replace its layer-26 image-token states at every step with those computed under another task's instruction B, and count how often the robot performs B (Appendix F).

Object-attention lift. A policy that grounds the instruction should direct its instruction attention toward the objects it names. Let a_i be the attention from instruction tokens to image token i of one camera, averaged over queries, heads and layers, and S the tokens covered by the segmentation of the task objects among the P image tokens. The lift

$$L \;=\; \frac{\sum_{i \in S} a_i \,/\, \sum_i a_i}{|S| \,/\, P} \tag{3}$$

divides the objects' share of the attention by their share of the image. L=1 means uniform attention and L=2 that the objects receive twice the attention their area predicts. We compute L for both cameras on 240 state-matched scenes of the 40 base tasks. For the reverse direction we report how much attention the image tokens pay to the instruction, and we also measure the action head's attention on the instruction tokens. Further probes are defined in Appendix D.

4 EXPERIMENTS

We ask four questions. **RQ1: Does bidirectional attention help?** Does it improve a VLA, and does it matter when the attention mask is learned (Section 4.2)? **RQ2: Where is the instruction represented?** Does a bidirectional attention mask move the instruction into the image tokens (Section 4.3)? **RQ3: Does the policy act on it?** Does the policy act on the instruction carried in its image tokens (Section 4.4)? **RQ4: Why does only a learned mask help?** What separates a learned attention mask from a flip (Section 4.5)?

Table 1: **Six paradigms from one causal Qwen3-VL-2B.** Success (%). LIBERO-Plus and RoboCasa365: mean of two seeds, per-seed values in brackets. P&P: the four pick-and-place tasks (grouped post hoc). Bridge: one seed, ± binomial SE. **Bold**: best, underlined: second.

	Configuration		LIBERO-Plus	RoboCasa365		Bridge	
	Pretrain.	Attn. mask	Zero-shot ↑	All ↑	P&P ↑	Stock ↑	Distr. ↑
P1	None	Causal	73.9 [73.2, 74.7]	67.9 [67.8, 68.0]	56.2 [56.0, 56.4]	45.0 ±2.3	45.8 ±2.9
P2	Causal	Causal	75.9 [75.9, 76.0]	68.2 [67.5, 68.9]	55.9 [53.4, 58.5]	59.4 ±2.2	49.3 ±2.9
P3	None	Flipped	71.9 [71.8, 72.1]	63.9 [62.5, 65.3]	58.4 [58.1, 58.6]	42.1 ±2.3	38.9 ±2.9
P4	Causal	Flipped	72.4 [70.8, 74.1]	60.0 [56.9, 63.0]	51.0 [47.6, 54.4]	53.1 ±2.3	43.8 ±2.9
P5	PrefixLM	Learned	**78.3** [77.9, 78.7]	68.6 [68.2, 68.9]	59.8 [56.8, 62.9]	46.2 ±2.3	45.1 ±2.9
P6	Enc.-Dec.	Learned	78.2 [77.4, 79.0]	**68.9** [68.1, 69.6]	**61.9** [57.9, 66.0]	**61.9** ±2.2	**56.9** ±2.9

4.1 Experimental setup

LIBERO-Plus. Policies train on the four LIBERO suites (Spatial, Object, Goal and Long, 40 tasks of human demonstrations) with an external and a wrist camera at 224×224, 7-d delta end-effector actions in chunks of 8, and 8 state tokens (raw proprioception clipped to $[-1, 1]$ and binned), for 30k steps at batch 32. They are evaluated zero-shot on all LIBERO-Plus (Fei et al., 2026), which perturb camera, initial state, instruction wording, lighting, background, sensor noise, and layout. We do not report standard LIBERO (Liu et al., 2023) results, as all models saturate above 96%.

RoboCasa365. Policies train on the official human demonstrations of the atomic tasks of RoboCasa365 (Nasiriany et al., 2026) (mobile PandaOmron arm in the target kitchens, two cameras at 256×256, 12-d arm and base actions in chunks of 16 of which 8 are executed, 32 state tokens, 50k steps at batch 64). We evaluate 17 atomic tasks (four pick-and-place, twelve articulation, one navigation), 200 episodes per task and seed, two seeds.

BridgeData V2 and SimplerEnv. To test policies trained on real images, all paradigms train on BridgeData V2 (WidowX, one third-person 224×224 camera, 7-d delta end-effector chunks of 16, 30k steps at batch 256, one seed). We evaluate in SimplerEnv WidowX on the four stock tasks (480 episodes) and on a distractor variant that adds two objects to each of the three table tasks (288 scenes fixed before any rollout, Appendix H).

Reasoning, external models, statistics. We score pretrained backbones on 9 embodied spatial-reasoning benchmarks (Appendix C) and probe the publicly released LIBERO checkpoints of two prominent bidirectional VLAs, converted to the StarVLA codebase (Community, 2026). All paradigms are evaluated on identical episodes. The text reports means over seeds. Intervals are paired 95% bootstraps over tasks. For groups of three or four tasks we report per-task values with episode-level z (Appendix B).

4.2 RQ1: Only a learned attention mask improves beyond causal pretraining

Bidirectional attention helps only when it is learned during pretraining. Learning a bidirectional mask during embodied pretraining gives the best policies, improving on the causal base by 4.4 and 4.3 points on LIBERO-Plus for PrefixLM and conversion, with every seed's 95% interval above zero (Table 1, Figure 6). Flipping the same attention mask at policy time without retraining under it makes the policy worse in every benchmark total, whether or not the backbone was pretrained. The gain does not come from the pretraining data alone: causal pretraining on the same corpus gains only half as much on LIBERO-Plus and nothing on RoboCasa365, and the learned masks outperform it at equal data. On Bridge, where the pretraining corpus covers the evaluation data, causal pretraining already gains strongly and conversion adds to it mainly among distractors.

The gains appear where the instruction must single out a task or an object. The learned masks gain most where the scene alone does not determine what to do. On LIBERO-Goal, whose tasks share one scene, they gain about 9 points over P1. On RoboCasa365, which is a tie overall, conversion gains 5.7 points on the four pick-and-place tasks, where an object and a destination must be chosen among several, and nothing on the articulation tasks, where the scene fixes the target. On

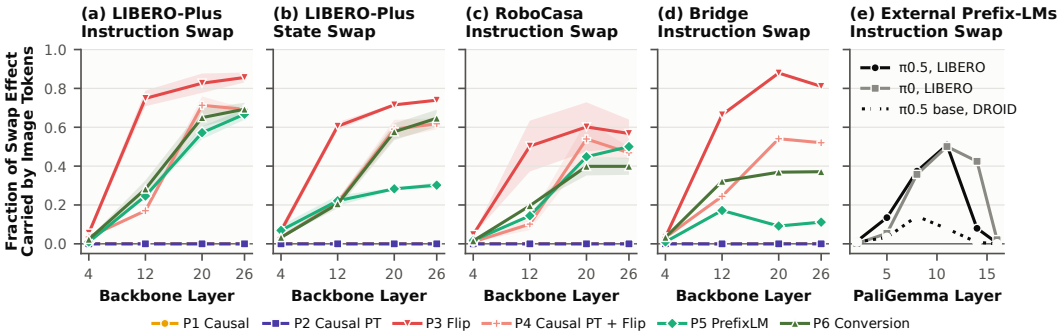


Figure 3: **Every bidirectional paradigm carries the instruction in its image tokens, no causal one does.** Share of a swapped instruction's (a, c–e) or state's (b) effect on the actions that patching all image-token states at one layer transmits. (a–c) Two seeds, bands span them. (d) One Bridge seed, 480 SimplerEnv frames, one third-person camera. Causal paradigms are zero by construction. (e) The π action expert reads every PaliGemma layer (18), so the drop at its final layer is by construction. Dotted: $\pi_{0.5}$ base model on DROID.

Bridge, conversion leads causal pretraining by 7.6 points among distractors, mostly on the task that asks for one of two blocks by colour (+18.8, Appendix H). The LIBERO-Plus perturbations point the same way: the gains come from rearranged objects and initial states, not from camera or lighting changes (Appendix E). The learned masks therefore improve how the instruction is tied to the right object, not visual robustness, which RQ2 to RQ4 examine at the level of representations.

4.3 RQ2: A BIDIRECTIONAL ATTENTION MASK MOVES THE INSTRUCTION INTO THE IMAGE TOKENS

Bidirectional image tokens absorb the instruction mid-stack. Activation patching copies the image-token states computed with a swapped instruction into the original run and measures how far the actions shift toward the swapped instruction (Figure 3a). In every bidirectional policy the instruction starts in its own tokens and is progressively written into the image tokens through the backbone layers. On LIBERO-Plus at layer 26, one layer before the head, the image tokens carry two thirds or more of its effect, mostly through the wrist view. Under a causal mask the share is zero by construction, and the instruction stays in its own tokens and those after it. The robot state enters the image tokens the same way (Figure 3b), consistent with the finding that a corrupted state hurts converted policies more than causal ones (Appendix E). π_0 and $\pi_{0.5}$ show the same pattern (Figure 3e). A bidirectional backbone therefore hands its action head image representations that already encode what the instruction asks for, whereas a causal backbone leaves this binding to later tokens.

A bidirectional attention mask opens the image route, and training shapes it. Flipping the attention mask of a trained causal policy at probe time, without any retraining, moves only about half of the instruction's effect into the image tokens, late and evenly across both cameras (Table 15). Training controls how much, how early, and through which camera the instruction enters. The flip without pretraining writes the instruction in early and heavily. The learned masks do so gradually. PrefixLM routes it almost entirely through the wrist view. With Bridge's single third-person camera these differences become stark: the image tokens carry four fifths of the instruction's effect for the untrained flip, half for the pretrained flip, a third for conversion, and a tenth for PrefixLM (Figure 3d, Table 23). This suggests why PrefixLM matches conversion in simulation but stays at the causal base level on Bridge (46.2 against 61.9% on stock scenes, Table 1): without a wrist camera, its image route barely forms. Carrying more of the instruction does not predict success, however. The untrained flip carries the most on every benchmark and is the weakest policy. What matters is not how much of the instruction reaches the image tokens but what form it takes, which RQ4 examines.

4.4 RQ3: THE POLICY ACTS ON THE INSTRUCTION IN THE IMAGE TOKENS

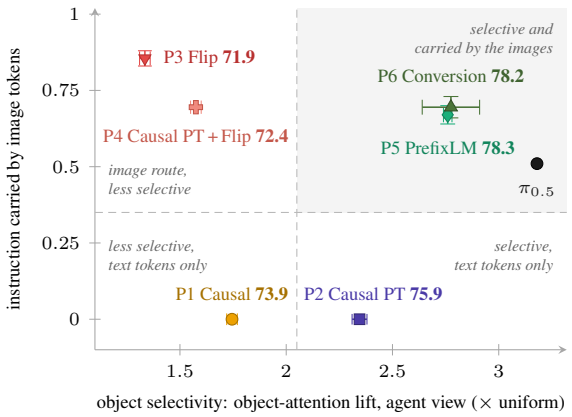


Figure 4: **Object selectivity and image-carried instruction.** Object-attention lift (x, agent view) against image-token instruction share at layer 26 (y). Bars span two seeds. Labels give LIBERO-Plus success (%). Causal image shares are zero by construction. Black point: an external bidirectional VLA, with its image share taken at the peak layer. Numerical values in Table 14.

The image tokens decide which task the policy performs. In a closed-loop transplant on LIBERO-Goal, the policy keeps its own instruction A, but at every step its image-token states are replaced by those computed with another task's instruction B (Table 2). The learned masks then perform task B in 82 and 78% of episodes and task A in none, and the flips in 70 and 49% (RQ4). The causal policy is unaffected, since its image tokens cannot depend on the instruction. The reverse knockout drops LIBERO-Goal to 6 to 22% in every bidirectional policy and costs 41 to 50 points on LIBERO-Plus and RoboCasa365 (Table 14). $\pi_{0.5}$ and π_0 depend on the same route, losing 30 and 60 points without it. Replacing the image tokens is thus enough to redirect the task in most episodes, and removing their route collapses LIBERO-Goal: in bidirectional policies, the visual representation is where the task is represented.

Table 2: **Instruction transplant** on LIBERO-Goal (success %, seed 42, seed 43 in brackets). Layer-26 image tokens from a pass with instruction B, instruction tokens keep A. Control: B's scene, instruction A, no transplant. Rev. KO: success when image tokens cannot read instruction and state (clean in brackets).

	clean	does B	does A	control	rev. KO
P1 causal	98	0	98	0	–
P3 flip	96.5	70	3	0	6 (68)
P4 PT+flip	98	49	2.5	0	19 (70)
P5 PrefixLM	99	82 (93)	0	0	10 (79)
P6 conversion	98.5	78 (85)	0	0	22 (76)

The transplanted image tokens, not the scene, carry the task. LIBERO is largely solvable from the scene alone (Fei et al., 2026; Zhou et al., 2025b). On plain LIBERO without any instruction, our causal base still solves 60% of Spatial and 58% of LIBERO-10, whereas the converted encoder falls to 20% and 7% (Table 12). We therefore test task identity on LIBERO-Goal, whose ten tasks share one scene and draw their initial states from the same distribution. Its matched control, B's scene with instruction A and no transplant, never yields task B (0 of 200), so B reaches the actions only through the transplanted image tokens. Because the transplant sets the two token groups against each other, it shows which one the head follows, not that the instruction tokens are unused, since on natural inputs they still carry 0.1 to 0.2 of the effect in the learned masks (Table 18). The causal policy can solve much of LIBERO from the scene and use the instruction only where the scene is ambiguous. The bidirectional policies instead write the instruction into their visual representation and act on it, on every suite: the reverse knockout costs them 27 to 55 points even where the scene determines the task.

4.5 RQ4: Why only a learned attention mask helps

Learned masks preserve object-selective attention, flips lose it. Figure 4 combines two probes: the object-attention lift (x), which measures how strongly the instruction's attention concentrates on the segmented task objects, and activation patching (y), which measures how much of an instruction swap's effect on the actions the image tokens carry. Causal models lie on the x-axis, since their image tokens cannot carry the instruction. The learned masks sit top right: they carry the instruction and attend to the task objects at 2.8 times the uniform rate. The flips carry as much but attend at only 1.3 to 1.6, below the causal models they were copied from. $\pi_{0.5}$, also trained with a learned mask, joins the learned masks (3.2). Embodied pretraining already raises the lift under a causal mask (1.7 to 2.3). Learning the mask during pretraining amplifies it on the agent view (2.8), and flipping the mask afterwards undoes it on both cameras.

Flips mix in more of the instruction and follow it less reliably. The flips' image tokens read the instruction with about twice the attention of the learned masks (0.09 against 0.05 per token), with the same mild preference for the task objects, and their action heads nearly stop attending to the instruction tokens (0.02 to 0.03 of their attention, against 0.12 to 0.17 for every other paradigm, Table 16 and Table 14). In the transplant they follow the transplanted task in 70 and 49% of episodes, against 82 and 78% for the learned masks (Table 2), and transplanting at the head's input lowers them further (55 and 45%) while the learned masks stay unchanged (Appendix F). The flips thus carry more of the instruction in their image tokens, but in a form the policy follows less reliably.

Summary. The four questions suggest a unified account. Embodied pretraining makes the instruction's attention object-selective, even under a causal mask. A bidirectional mask carries the instruction into the image tokens (RQ2), from which the policy reads its task (RQ3). Learned during pretraining, the mask keeps this selectivity (lift 2.8), and flipped afterwards, it loses it (1.6). The gains therefore appear where the instruction must single out an object (RQ1). The pretraining objective also sets how strongly the instruction enters the image tokens. The encoder's decoder reads only its final layer, which forces fusion into the image tokens, whereas PrefixLM can keep the instruction in its text tokens and edits its image tokens only half as much, mostly in the wrist view. We hypothesize that this is why it fails on Bridge, which has no wrist camera.

5 Discussion, Limitations, and Conclusion

Learning the attention mask, not the mask itself, is what helps. Policy training alone does not restore the object-selective instruction attention that an attention-mask flip loses (Figure 4). As in language models, where enabling bidirectional attention degrades most causal models until a dedicated adaptation stage (BehnamGhader et al., 2024), studies that impose the attention mask only at policy time (Ding et al., 2026; Chen et al., 2026) may underestimate bidirectional attention. Our results suggest learning the mask during embodied pretraining rather than imposing it at policy time. Conversion additionally improves reasoning (Appendix C) and real-image transfer, though it requires a second text stack during pretraining.

The head can supply the fusion instead. The conversion gain depends on the action head (Table 24). With the final-layer GR00T-style head, conversion improves LIBERO-Plus by 3.0 points (seed 42, normalised state). With a layer-wise head (Appendix I) the gap closes from the causal side: the causal policy rises to the encoder's level (78.1 against 78.9%), and on RoboCasa365 pick-and-place it surpasses it. The encoder still carries the instruction in its image tokens under the layer-wise head (0.44 to 0.47 at layer 20), so the gain disappears because the causal policy catches up, not because the encoder loses its route. One reading is that a policy needs image-instruction fusion once, either in the image tokens of a bidirectional backbone or in a layer-wise head that attends to every layer. The head interaction is not significant ($+2.1$ $[-0.6, +4.8]$), so this remains a hypothesis.

Limitations. All policies are evaluated in simulation. The ladder uses one 2B VLM family, and a one-seed Qwen3.5-0.8B control reverses the result (conversion 72.3%, causal 75.6%). With normalised state tokens the conversion gain is 3.0 points at one seed (Appendix I). The learned masks' selectivity advantage holds on the agent view only, and each pretraining variant is a single run. The RoboCasa365 grouping is post hoc, and the Bridge lead rests on one seed and mostly one task.

Conclusion. Using six VLA paradigms built from the same initial backbone and evaluated with matched policy data and action heads, we showed that a bidirectional attention mask moves the instruction into the image tokens, that the policy reads its task from there, and that only a mask learned during embodied pretraining keeps the instruction's attention object-selective and improves manipulation. An attention-mask flip opens the same route but loses this selectivity, and policy training does not recover it. Our results favor learning the bidirectional mask during embodied pretraining when using a final-layer action head.

AI use statement

In this work, we used generative AI tools (Anthropic's Claude models) to implement methods and to provide feedback on experiments. We have not used generative AI tools to generate synthetic data sets, to develop theoretical models or conceptual frameworks, to formulate mathematical claims, to

propose or refine hypotheses, to clean or reformat data sets, or to interpret results. Proofs, translation, and qualitative or thematic data analysis are not applicable to this work.

Additionally, we used generative AI tools to create and debug software code, including figure plotting code, to search for and identify relevant literature, to format references, to draft parts of the research paper and to edit the paper for readability.

We have reviewed all AI-assisted work. In particular, the authors reviewed generated code and unit tests, traced every number reported in the paper to a logged evaluation run, and checked every citation against its published version.

We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

Ethics statement

This work uses only simulated robot benchmarks and publicly available datasets; it involves no human subjects, personal data or deployment on physical hardware.

References

Xiang An, Yin Xie, Feilong Tang, Yunyao Yan, Huajie Tan, Didi Zhu, Chang-Rui Chen, Xiuwei Zhao, Bin Qin, Kaicheng Yang, Yifei Shen, Yuanhan Zhang, Kai-Chen Zhang, Wenkang Zhang, Zheng-Xue Cheng, Nan Zhang, C. Wu, Chunjiang Ge, Zimin Ran, Dehua Song, Chunyuan Li, Shikun Feng, Mingxue Hu, Zhangquan Chen, Junbo Niu, Bo Li, Zi-Yong Feng, Ziwei Liu, Zongyuan Ge, and Jiankang Deng. LLaVA-OneVision-2: Towards next-generation perceptual intelligence. *arXiv.org*, 2026. doi: 10.48550/arXiv.2605.25979.

Shuai Bai, Yuxuan Cai, Rui-Zhe Chen, Ke-qin Chen, Xiong-Hui Chen, Zesen Cheng, Liang-Hao Deng, Wei Ding, Rongyao Fang, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Qidong Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Ming-Sheng Li, Mei Li, Kaixin Li, Zi-Cheng Lin, Junyang Lin, Xue-Jing Liu, Jiawei Liu, Cheng-Long Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chen-Xu Lv, Rui Men, Lingchen Meng, Lingchen Meng, Xin-yi Ren, Sibo Song, Yu-chen Sun, Jun Tang, Jianhong Tu, Jian-Qiang Wan, Peng Wang, Pengfei Wang, Qiu-Yue Wang, Yuxuan Wang, Tian-Bao Xie, Yihe Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bo-Wen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Botao Zheng, Humen Zhong, Jingren Zhou, Fanxi Zhou, Jingren Zhou, Yuanzhi Zhu, and Keming Zhu. Qwen3-VL technical report. *arXiv.org*, 2025. doi: 10.48550/arXiv.2511.21631.

Parishad BehnamGhader, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados, and Siva Reddy. LLM2Vec: Large language models are secretly powerful text encoders. In *arXiv.org*, 2024. doi: 10.48550/arXiv.2404.05961.

L. Beyer, A. Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel M. Salz, Maxim Neumann, Ibrahim M. Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, Thomas Unterthiner, Daniel Keysers, Skanda Koppula, Fangyu Liu, Adam Grycner, A. Gritsenko, N. Houlsby, Manoj Kumar, Keran Rong, Julian Martin Eisenschlos, Rishabh Kabra, Matthias Bauer, Matko Bosnjak, Xi Chen, M. Minderer, P. Voigtlaender, Ioana Bica, Ivana Balazevic, J. Puigcerver, Pinelopi Papalampidi, O. Hénaff, Xi Xiong, Radu Soricut, Jeremiah Harmsen, and Xiao-Qi Zhai. PaliGemma: A versatile 3B VLM for transfer. *arXiv.org*, 2024. doi: 10.48550/arXiv.2407.07726.

Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Lin-Xi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, J. Jang, Zhenyuan Jiang, Jan Kautz, Kaushil Kundalia, Lawrence Lao, Zhiqi Li, Zongyu Lin, Kevin Lin, Guilin Liu, Edith Llontop, Loic Magne, A. Mandlekar, Avnish Narayan, Soroush Nasiriany, Scott Reed, You Tan, Guanzhi Wang, Zu Wang, Jing Wang, Qi Wang, Jiannan Xiang, Yuqi Xie, Yinzhen Xu, Zhen-Teng Xu, Seonghyeon Ye, Zhiding Yu, Ao Zhang, Hao Zhang, Yizhou Zhao, Ruijie Zheng, and Yuke Zhu. GR00T N1: An open foundation model for generalist humanoid robots. *arXiv.org*, 2025. doi: 10.48550/arXiv.2503.14734.

Kevin Black, Noah Brown, Danny Driess, A. Esmail, M. Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, S. Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, L. Shi, James Tanner, Quan Vuong, Anna Walling, Hao-Huan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control. In *Robotics: Science and Systems XXI*. Robotics: Science and Systems Foundation, 2025. doi: 10.15607/RSS.2025.XXI.010.

Nils Blank, Paul Mattes, Maximilian Xiling Li, Jakub Suliga, Thomas Roth, Moritz Reuss, Pankhuri Vanjani, and Rudolf Lioutikov. SPARC: Reliable spatial annotations from robot demonstrations at scale. In *arXiv (Cornell University)*. Cornell University, 2026. doi: 10.48550/arxiv.2606.13497. arXiv:2606.13497.

Zhongang Cai, Rui-Si Wang, Chenyang Gu, Fanyi Pu, Junxiang Xu, Yubo Wang, Wanqi Yin, Zhitao Yang, Chen Wei, Qingping Sun, Tongxi Zhou, Jiaqi Li, Hui Pang, O. Qian, Yu-Rou Wei, Zhiqian Lin, Xuanke Shi, Kewang Deng, Xiaoyang Han, Zukai Chen, Xiangyu Fan, Hanming Deng, Lewei Lu, Liang Pan, Bo Li, Ziwei Liu, Quan Wang, Dahua Lin, and Lei Yang. Scaling spatial intelligence with multimodal foundation models. In *arXiv.org*, 2026. doi: 10.48550/arXiv.2511.13719.

Jiayi Chen, Wenxuan Song, Pengxiang Ding, Ziyang Zhou, Han Zhao, Feilong Tang, Donglin Wang, and Haoang Li. Unified diffusion VLA: Vision-language-action model via joint discrete denoising diffusion process. In *arXiv.org*, 2026. doi: 10.48550/arXiv.2511.01718.

Christopher Clark, Jieyu Zhang, Zixian Ma, J. Park, Mohammadreza Salehi, Rohun Tripathi, Sangho Lee, Zhongzheng Ren, Chris Dongjoo Kim, Yinuo Yang, Vincent Shao, Yue Yang, Weikai Huang, Ziqi Gao, Taira Anderson, Jianrui Zhang, Jitesh Jain, George Stoica, Winson Han, Ali Farhadi, and Ranjay Krishna. Molmo2: Open weights and data for vision-language models with video understanding and grounding. In *arXiv.org*, 2026. doi: 10.48550/arXiv.2601.10611.

S. Community. StarVLA: A lego-like codebase for vision-language-action model developing. arXiv preprint arXiv:2604.05014, 2026.

Zeyuan Ding, Wenhai Liu, Yang Xu, Jiayu Hu, Yinda Chen, Yi Zhang, Yong Dai, Jian Tang, and Xiaozhu Ju. Pelican-VLA 0.5: Attending before acting benefits generalization. *arXiv.org*, 2026. doi: 10.48550/arXiv.2607.06655.

Haoquan Fang, Jiafei Duan, D. Clay, Sam Wang, Shuo Liu, Weikai Huang, Xiang Fan, Wei-Chuan Tsai, Shirui Chen, Y. Wang, Shanli Xing, Jaemin Cho, J. Park, Ainaz Eftekhar, Petr Sushko, Karen Farley, Angad Wadhwa, Col Harrison, Winson Han, Ying-Chun Lee, Eli VanderBilt, Rose Hendrix, Suveen Ellawela, Lucas Ngoo, Joyce Chai, Zhongzheng Ren, Ali Farhadi, Dieter Fox, and Ranjay Krishna. MolmoAct2: Action reasoning models for real-world deployment. *arXiv.org*, 2026. doi: 10.48550/arXiv.2605.02881.

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: A progressive robustness benchmark for visual-language-action models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 38574–38583, 2026.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. Dissecting recall of factual associations in auto-regressive language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12216–12235. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.751.

Jisang Han, Seonghu Jeon, Jaewoo Jung, René Zurbrügg, Honggyu An, Tifanny Portela, Marco Hutter, Marc Pollefeys, Seungryong Kim, and Sung-Jin Hong. Geometric action model for robot policy learning. *arXiv.org*, 2026. doi: 10.48550/arXiv.2606.17046.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, A. Esmail, M. Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, S. Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, L. Shi, Laura Smith, J. Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, H. Walke, Anna

Walling, Hao-Huan Wang, Li-Li Yu, and Ury Zhilinsky. $\pi_{0.5}$: a vision-language-action model with open-world generalization. In *arXiv.org*, volume 305 of *Proceedings of Machine Learning Research*, pp. 17–40, 2025. doi: 10.48550/arXiv.2504.16054.

Yuheng Ji, Huajie Tan, Jiayu Shi, Xiaoshuai Hao, Yuan Zhang, Hengyuan Zhang, Pengwei Wang, Mengdi Zhao, Yao Mu, Pengju An, Xinda Xue, Qinghang Su, Huaihai Lyu, Xiaolong Zheng, Jiaming Liu, Zhongyuan Wang, and Shanghang Zhang. RoboBrain: A unified brain model for robotic manipulation from abstract to concrete. In *Computer Vision and Pattern Recognition*, pp. 1724–1734. IEEE, 2025. doi: 10.1109/CVPR52734.2025.00168.

Nikita Kachaev, Mikhail Kolosov, Daniil Zelezetsky, Alexey K. Kovalev, and Aleksandr I. Panov. Don't blind your VLA: Aligning visual representations for OOD generalization. In *Proceedings of the 25th International Conference on Autonomous Agents and Multiagent Systems*, pp. 1864. International Foundation for Autonomous Agents and Multiagent Systems, 2025. doi: 10.65109/PPER9186.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An open-source vision-language-action model. In *Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 2679–2713, 2024. doi: 10.48550/arXiv.2406.09246.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success. In *Robotics: Science and Systems XXI*. Robotics: Science and Systems Foundation, 2025. doi: 10.15607/RSS.2025.XXI.017.

Jung Min Lee, Dohyeok Lee, Seokhun Ju, Taehyun Cho, Jin Woo Koo, Li Zhao, Sangwoo Hong, and Jungwoo Lee. MVP-LAM: Learning action-centric latent action via cross-viewpoint reconstruction. In *arXiv (Cornell University)*. Cornell University, 2026. doi: 10.48550/arxiv.2602.03668.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. LLaVA-OneVision: Easy visual task transfer. *Trans. Mach. Learn. Res.*, 2025. doi: 10.48550/arXiv.2408.03326.

Hao Li, Ziqin Wang, Zi-han Ding, Shuai Yang, Yilun Chen, Yang Tian, Xiaolin Hu, Tai Wang, Dahua Lin, Feng Zhao, Si Liu, and Jiangmiao Pang. RoboInter: A holistic intermediate representation suite towards robotic manipulation. In *arXiv.org*, 2026. doi: 10.48550/arXiv.2602.09973.

Xinghang Li, Peiyan Li, Long Qian, Minghuan Liu, Dong Wang, Jirong Liu, Bingyi Kang, Xiao Ma, Xinlong Wang, Di Guo, Tao Kong, Hanbo Zhang, and Huaping Liu. What matters in building vision–language–action models for generalist robots. *Nature Machine Intelligence*, 8(2):158–172, 2024a. doi: 10.1038/s42256-025-01168-7.

Xuanlin Li, Kyle Hsu, Jiayuan Gu, Karl Pertsch, Oier Mees, Homer Rich Walke, Chuyuan Fu, Ishikaa Lunawat, Isabel Sieh, Sean Kirmani, Sergey Levine, Jiajun Wu, Chelsea Finn, Hao Su, Quan Vuong, and Ted Xiao. Evaluating real-world robot manipulation policies in simulation. *Conference on Robot Learning*, 2024b. doi: 10.48550/arXiv.2405.05941.

Weifeng Lin, Siyuan Huang, Hao Li, Tingwei Chen, Ruichuan An, Xinyu Wei, Jianbo Liu, and Hongsheng Li. Rethinking VLM representation for VLA initialization. *arXiv (Cornell University)*, 2026. doi: 10.48550/arxiv.2605.25802.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems 36*, pp. 44776–44791, 2023. doi: 10.52202/075280-1939.

Zhe Liu, Jinghua Hou, Yuxiang Lu, Zhenya Yang, Xianzhe Fan, Junwei Luo, Junyi Li, Ruihua Han, Zhi Hou, and Hengshuang Zhao. StreamPI: Streaming multimodal temporal modeling for vision-language-action models. *arXiv*, 2026.

Yuhao Lu, Yixuan Fan, Beixing Deng, Fangfu Liu, Yali Li, and Shengjin Wang. VL-Grasp: a 6-dof interactive grasp policy for language-oriented objects in cluttered indoor scenes. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 976–983. IEEE, 2023. doi: 10.1109/iros55552.2023.10341379.

Qi Lv, Weijie Kong, Hao Li, Jia Zeng, Zherui Qiu, Delin Qu, Haoming Song, Qizhi Chen, Xiang Deng, and Jiangmiao Pang. F1: A vision-language-action model bridging understanding and generation to actions. *arXiv.org*, 2025. doi: 10.48550/arXiv.2509.06951.

Haoxiang Ma, Junhao Cai, Xiaoxu Xu, Hao Li, Yu-Yin Yang, Yang Tian, Jia-Fei Cao, Hong-Rui Zhu, Zherui Qiu, Zhaxizhuoma, Yu-Qiang Yang, Jiaqi Peng, Xueyuan Wei, Yang-Kun Zhu, Jiahao Jiang, Xin-Bo Gao, Han-Qing Wang, Feng Yuan, Kailin Li, Xueyue Zhu, Tai Wang, Yan Ding, Jiang-Miao Pang, Jia Zeng, Jing-Jing Zhang, Bowen Zhou, Yao Mu, Chunhua Shen, and Weinan Zhang. InternVLA-A1.5: Unifying understanding, latent foresight, and action for compositional generalization. *arXiv.org*, 2026. doi: 10.48550/arXiv.2607.04988.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Neural Information Processing Systems*, pp. 17359–17372. Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2022. doi: 10.52202/068431-1262.

Soroush Nasiriany, Sepehr Nasiriany, Abhiram Maddukuri, and Yuke Zhu. RoboCasa365: A large-scale simulation framework for training and benchmarking generalist robots. In *arXiv.org*, 2026. doi: 10.48550/arXiv.2603.04356.

Delin Qu, Haoming Song, Chen Qizhi, Zhaoqing Chen, Xianqiang Gao, Dong Wang, Xiao-Wei Ye, Qi Lv, Modi Shi, Guanghui Ren, Cheng Ruan, Yao Maoqing, Haoran Yang, Jiacheng Bao, Bin Zhao, and Li Xuelong. EO-1: An open unified embodied foundation model for general robot control. *arXiv (Cornell University)*, 2025. doi: 10.48550/arxiv.2508.21112.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2019.

Moritz Reuss, Hongyi Zhou, Marcel Rühle, Ömer Erdinç Yağmurlu, Fabian Otto, and Rudolf Lioutikov. FLOWER: Democratizing generalist robot policies with efficient vision-language-flow models. In *Proceedings of the 9th Conference on Robot Learning (CoRL)*, volume 305 of *Proceedings of Machine Learning Research*, pp. 3736–3761, 2025.

Fei Senyu, Wang Siyin, Ji Li, Li Ao, Zhang Shiduo, Liu Liming, Hou Jinlong, Gong Jingjing, Zhao Xianzhong, and Qiu Xipeng. SRPO: Self-referential policy optimization for vision-language-action models. In *arXiv (Cornell University)*, pp. 6718–6728. Cornell University, 2026. doi: 10.48550/arxiv.2511.15605.

Chan Hee Song, Valts Blukis, Jonathan Tremblay, Stephen Tyree, Yu Su, and Stan Birchfield. RoboSpatial: Teaching spatial understanding to 2D and 3D vision-language models for robotics. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15768–15780. IEEE, 2025. doi: 10.1109/cvpr52734.2025.01470.

Yi Tay, Mostafa Dehghani, Vinh Q. Tran, Xavier Garcia, Jason Wei, Xuezhi Wang, Hyung Won Chung, Siamak Shakeri, Dara Bahri, Tal Schuster, Huaixiu Steven Zheng, Denny Zhou, Neil Houlsby, and Donald Metzler. UL2: Unifying language learning paradigms. In *International Conference on Learning Representations*, 2022.

Fangyuan Wang, Peng Zhou, Jiaming Qi, Shipeng Lyu, Chengyang He, David Navarro-Alarcon, and Guodong Guo. Think proprioceptively: State-grounded visual token selection for VLA policies. *arXiv*, 2026a.

Runze Wang, Yuqian Fu, Yu Li, Tao Lin, Tianwen Qian, Mohamed Elhoseiny, Bo Zhao, Yanwei Fu, Yu-Gang Jiang, and Xiangyang Xue. Afford-VLA: Action-aligned visual planning via internalized affordance. *arXiv.org*, 2026b. doi: 10.48550/arXiv.2605.24203.

Xiao-Ming Wu, Bin Fan, Kang Liao, Jian-Jian Jiang, Runze Yang, Yihang Luo, Zhonghua Wu, Wei-Shi Zheng, and Chen Change Loy. VLANeXt: Recipes for building strong VLA models. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*, 2026.

Yifu Xiong, Wenhao Yu, Jiaxuan Lin, Bojun Zou, Jiahao Li, Lu Zhang, Yanyong Zhang, and Jianmin Ji. CAC-VLA: Context-gated action conditioning for vision-language-action models. *arXiv.org*, 2026. doi: 10.48550/arXiv.2607.04816.

Wentao Yuan, Jiafei Duan, Valts Blukis, Wilbert Pumacay, Ranjay Krishna, Adithyavairavan Murali, Arsalan Mousavian, and Dieter Fox. RoboPoint: A vision-language model for spatial affordance prediction in robotics. In *Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 4005–4020, 2024.

Yifu Yuan, Haiqin Cui, Yibin Chen, Zibin Dong, Fei Ni, Longxin Kou, Jinyi Liu, Pengyi Li, Yan Zheng, and Jianye Hao. From seeing to doing: Bridging reasoning and decision for robotic manipulation. In *arXiv.org*, 2026a. doi: 10.48550/arXiv.2505.08548.

Yifu Yuan, Yao-Bang Huang, X. Yao, Yutong Li, Shuo Zhang, Linqi Han, Peng-Yi Li, Jiangeng Sun, Wenting Jia, Zhao-Xiang Zhang, Yuhao Liu, Ruihao Liao, Yucheng Hu, Qiyu Wu, Yuxiao Li, Zibin Dong, Fei Ni, Yangheng Zheng, Shuyang Gu, Yi Ma, Hongyao Tang, Han Hu, and Jianye Hao. Embodied-R1.5: Evolving physical intelligence via embodied foundation models. *arXiv.org*, 2026b. doi: 10.48550/arXiv.2606.11324.

Biao Zhang, Fédor Moiseev, Joshua Ainslie, Paul Suganthan G.C., Min Ma, Surya Bhupatiraju, Federico Lebrón, Orhan Fırat, Armand Joulin, and Zhe Dong. Encoder-decoder Gemma: Improving the quality-efficiency trade-off via adaptation. *arXiv (Cornell University)*, 2025a. doi: 10.48550/arxiv.2504.06225.

Biao Zhang, P. Suganthan, Gaël Liu, Ilya Philippov, Sahil Dua, B. Hora, Kat Black, Gus Martins, Omar Sanseviero, Shreya Pathak, Cassidy Hardin, Francesco Visin, Jia-Geng Zhang, Kathleen Kenealy, Qin Yin, Xiao-Dan Song, Olivier Lacombe, Armand Joulin, T. Warkentin, and Adam Roberts. T5Gemma 2: Seeing, reading, and understanding longer. *arXiv.org*, 2025b. doi: 10.48550/arXiv.2512.14856.

Chuheng Zhang, Rushuai Yang, Xiaoyu Chen, Kaixin Wang, Li Zhao, Yi Chen, and Jiang Bian. How do VLAs effectively inherit from VLMs? *arXiv.org*, 2025c. doi: 10.48550/arXiv.2511.06619.

Fred Zhang and Neel Nanda. Towards best practices of activation patching in language models: Metrics and methods. In *International Conference on Learning Representations*, 2024. doi: 10.48550/arXiv.2309.16042.

Jianke Zhang, Yanjiang Guo, Yucheng Hu, Xiaoyu Chen, Xiang Zhu, and Jianyu Chen. UP-VLA: A unified understanding and prediction model for embodied agent. In *International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 74911–74922, 2025d. doi: 10.48550/arXiv.2501.18867.

Jianke Zhang, Xiaoyu Chen, Yanjiang Guo, Yucheng Hu, and Jianyu Chen. VLM4VLA: Revisiting vision-language-models in vision-language-action models. In *International Conference on Learning Representations (ICLR)*, 2026a.

Zhewei Zhang, Puyue Wang, Guanren Qiao, Yijie Weng, Jiawei Hu, Guo Li, Lujia Wang, Junyan Wang, Tao Gu, Hongliang Lu, Guiliang Liu, Hong Jia, and Xinhu Zheng. LIRA: Local cross-layer information routing for vision-language-action decoding. *arXiv*, 2026b.

Jinliang Zheng, Jianxiong Li, Zhihao Wang, Dongxiu Liu, Xirui Kang, Yuchun Feng, Yinan Zheng, Jiayin Zou, Yilun Chen, Jia Zeng, Ya-Qin Zhang, Jiangmiao Pang, Jingjing Liu, Tai Wang, and Xianyuan Zhan. X-VLA: Soft-prompted transformer as scalable cross-embodiment vision-language-action model. In *arXiv.org*, 2026. doi: 10.48550/arXiv.2510.10274.

Enshen Zhou, Jingkun An, Cheng Chi, Yi Han, Shanyu Rong, Chi Zhang, Pengwei Wang, Zhongyuan Wang, Tiejun Huang, Lu Sheng, and Shanghang Zhang. RoboRefer: Towards spatial referring with reasoning in vision-language models for robotics. In *Advances in Neural Information Processing Systems 38*, pp. 32209–32286. Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2025a. doi: 10.52202/085713-0957.

Xueyang Zhou, Y. P. Xu, Guiyao Tie, Chen Yongchao, Guowen Zhang, Dong Chu, Pan Zhou, and Lichao Sun. LIBERO-PRO: Towards robust and fair evaluation of vision-language-action models beyond memorization. *arXiv (Cornell University)*, 2025b. doi: 10.48550/arxiv.2510.03827.

A TRAINING SETUP

Table 3: **Training configuration per benchmark.** All paradigms of one benchmark share every entry of its row. Common to all rows: AdamW (β=0.9, 0.95), cosine decay to 10^{-6} after a 1k-step warm-up, gradient clipping 1.0, crop and photometric augmentation, learning rates 10^{-5} (backbone), 10^{-4} (action head) and 2.5×10^{-5} (other modules), GR00T-style flow-matching DiT head reading the backbone's final layer, and 256-bin state tokens spliced after the instruction. The LIBERO and RoboCasa365 heads have 12 layers of width 768; the Bridge head is the 16-layer, width-1024 SimplerEnv reference head.

benchmark	cameras	resolution	actions	chunk	batch	steps	state tokens	seeds	paradigms
LIBERO	agent view + wrist	224×224	7-d delta EEF	8	32	30k	8 (clipped to $[-1, 1]$)	42, 43	P1–P6
RoboCasa365	left agent view + wrist	256×256	12-d arm + base	16 (execute 8)	64	50k	32 (normalised)	4, 42	P1–P6
Bridge	one third-person view	224×224	7-d delta EEF	16	256	30k	8 (q99-normalised)	42	P1, P2, P4, P5, P6

Actions and state. LIBERO actions are the 7-d end-effector deltas of the simulator's operational-space controller (position, rotation, gripper). The RoboCasa365 action concatenates end-effector position (3), rotation (3), gripper (1), base motion (4) and a control-mode flag (1). The LIBERO state tokens quantise eight proprioceptive values (position, roll, pitch, yaw, two finger joints) into 256 bins after clipping to $[-1, 1]$, identically for every paradigm.

Initialisation per paradigm. P1 and P3 start from the released Qwen3-VL-2B weights; P2 and P4 from the causally pretrained checkpoint (one epoch of the Embodied-Finetuning Mixture, Appendix C); P5 from the single-stack PrefixLM checkpoint on the same data; and P6 from the encoder of the converted encoder-decoder. P3, P4 and P6 run a bidirectional copy of the language stack and skip the decoder copy at policy time; P1, P2 and P5 keep a single stack.

Matched trainable parameters. The full language stack trains in every paradigm, and the executed trainable set is the same in all six: 2127.5M backbone parameters, 2244.3M with the action head. The paradigms with a copied encoder (P3, P4, P6) also instantiate the discarded decoder, so their logged trainable count includes its 352.5M cross-attention adapters; these never enter the policy's forward pass, and all 112 adapter tensors of a P6 checkpoint are bitwise unchanged after training.

Compute. Each LIBERO and RoboCasa365 policy trains on one node with four NVIDIA GH200 GPUs; each Bridge policy trains on four such nodes (16 GPUs). The embodied pretraining runs on 8 or 16 GH200 GPUs (Appendix C).

B EVALUATION PROTOCOL AND STATISTICS

LIBERO-Plus. Every LIBERO-Plus success rate evaluates all 10,030 perturbed instances once per policy, with the final 30k checkpoint and no checkpoint selection. The attention-path knockouts, which need many conditions per policy, roll out a fixed subset of 1,000 evenly spaced instances per suite (4,000 episodes), identical for every policy and condition. Plain LIBERO is near saturation for policies of this kind (above 95% for every configuration we evaluated on it), which is why LIBERO-Plus is the primary benchmark.

RoboCasa365 tasks and grouping. Policies train on the human demonstrations of all 18 atomic tasks of the RoboCasa365 target split and are evaluated on 17 of them, excluding `PickPlaceSinkToCounter`. Each task is evaluated with 200 clean episodes per training seed for success rates, and with 48 episodes per seed for the knockouts and state corruptions; episodes run for at most 500 control steps and execute 8 of the 16 predicted actions per chunk. We group the tasks by name after seeing the results: *pick-and-place* (P&P) are the four `PickPlace*` tasks (`CounterToCabinet`, `CounterToStove`, `DrawerToCounter`, `ToasterToCounter`); *articulation and appliances* are `CloseBlenderLid`, `CloseFridge`, `CloseToasterOvenDoor`, `CoffeeSetupMug`, `OpenCabinet`, `OpenDrawer`, `OpenStandMixerHead`, `SlideDishwasherRack`, `TurnOffStove`,

`TurnOnElectricKettle`, `TurnOnMicrowave` and `TurnOnSinkFaucet`; *navigation* is `NavigateKitchen`. Because `CoffeeSetupMug` also places an object, we checked the grouping by adding it to P&P: every paradigm's difference to P1 moves by less than one point and no sign changes, so the pick-and-place result does not hinge on this choice. The official 50-task RoboCasa365 protocol is not used, so our numbers are not comparable to its leaderboard (Appendix J).

π_0 and $\pi_{0.5}$. We use the openpi π_0 and $\pi_{0.5}$ LIBERO checkpoints (Black et al., 2025; Intelligence et al., 2025), converted to the StarVLA codebase (Community, 2026), on plain LIBERO (not LIBERO-Plus) with 5 trials on each of the 40 tasks, 200 episodes per condition (198 and 194 for the π_0 reverse-path and both-path knockouts). Clean success is 97.0% for $\pi_{0.5}$, matching the openpi release, and 90.5% for π_0. This $\pi_{0.5}$ checkpoint does not read the robot state, so it has no state probes.

Statistics. All paradigms of one benchmark are evaluated on identical episodes, and every comparison is paired by episode. The unit that is resampled depends on how many tasks a comparison spans. *Paired differences over a whole benchmark* (Figure 6, Table 20, the pooled two-seed LIBERO-Plus contrasts of Section 4.2, and the head interaction of Appendix I) use 95% bootstrap intervals that resample the 40 LIBERO base tasks or the 17 RoboCasa365 tasks with replacement and keep each task's paired episodes, of all seeds involved, together, because episodes of one task share scene, objects and instruction. The attention lift and the patching shares are likewise bootstrapped over base tasks. *Groups of three or four tasks* (the RoboCasa365 pick-and-place group and the three Bridge table tasks) are too few for a task bootstrap, whose resampling distribution would take only a handful of distinct values; for them we report per-task values with a binomial standard error over episodes and the paired z statistic, and state explicitly that task-level uncertainty cannot be estimated. These episode-level errors ignore within-task correlation and therefore understate the uncertainty. The same episode-level convention applies to all Bridge results (one training seed), whose image-share differences are bootstrapped over episodes. *Seeds*: Table 1 reports the mean over the two training seeds with both seed values in brackets; single-seed results are marked as such in their captions.

C Embodied pretraining and reasoning benchmarks

Pretraining setup. The three pretrained backbones (P2, P5, P6) see identical data and optimisation and differ only in the attention mask and, for P6, the second text stack. Each trains for one epoch of the Embodied-Finetuning Mixture (Table 4) with next-token prediction on the answer tokens, AdamW at learning rate 3×10^{-5} with 3% warm-up and cosine decay to 10%, no weight decay, gradient clipping at 1.0, sequences of at most 4,096 tokens, and length-grouped dynamic batches of 147,456 tokens per step (16 GH200 at 9,216 tokens for P6, 8 at 18,432 for P2 and P5), seed 42, FlashAttention-3 and bf16. Each pretraining variant is a single run with this seed, so the contrasts between P2, P5 and P6 include pretraining-run variance that the two policy seeds do not capture. The vision encoder (382M parameters) is frozen and its merger trained; every text layer trains. The encoder-decoder deep-copies all 28 text layers (1,409M each) into a bidirectional encoder and a causal decoder that attends to the encoder through the merged self- and cross-attention of T5Gemma 2 (Zhang et al., 2025b), sharing the vision tower and the embedding table (3.5B unique parameters in total).

Mixture. The Embodied-Finetuning Mixture combines 38 public datasets from eight releases (Table 4). Because the decoder is discarded after pretraining, we weight the data by how strongly the answer forces the visual representation: coordinate and dense outputs (points, boxes, traces), which cannot be produced from a language prior, receive the largest share (43.5%), followed by geometry (metric size, relations, correspondence; 26.7%), free-text robot QA and failure reasoning, which a strong decoder can answer largely from priors (13.9%), general VQA (12.8%). Caps subsample each source after shuffling rather than taking its head.

Reasoning benchmarks. Table 5 lists the twelve benchmark cells behind the reasoning aggregate, which is their unweighted mean. Several training sources share an origin with an evaluation benchmark (RoboRefIt, RefSpatial, RoboPoint for Where2Place); all pretrained backbones see the same data, so this affects every compared backbone equally.

Table 4: **The Embodied-Finetuning Mixture**, aggregated by data release (rows used after the per-source caps; share of the 6.72M rows).

source	content	rows used	share (%)
Embodied-R1.5 release (Yuan et al., 2026b)	25 sets: object and part grounding, pointing, grasp and contact points, visual traces, robot QA and failure reasoning (e.g. PACO-LVIS, PixMo-Points, RoboRefIt, RoboVQA, BridgeData failures)	2,875,832	42.8
Molmo2-ER (Clark et al., 2026; Fang et al., 2026)	spatial and multi-view reasoning (VSI, SAT, VST)	932,097	13.9
LLaVA-OneVision (Li et al., 2025)	general VQA and OCR	860,370	12.8
SenseNova-SI (Cai et al., 2026)	spatial intelligence: metric size, distance, relations	832,132	12.4
LLaVA-OneVision-2 spatial (An et al., 2026)	multi-image spatial and referring QA	678,546	10.1
EO-Data1.5M (Qu et al., 2025)	object referring, point and trajectory QA	262,051	3.9
SPARC robot frames (Blank et al., 2026)	object grounding, placement points and 8-point trajectories on OXE, BridgeData, DROID and AgiBot World frames	210,502	3.1
RoboPoint (Yuan et al., 2024)	spatial affordance points	53,412	0.8
FSD (Yuan et al., 2026a)	spatial relations and visual traces	12,221	0.2
Total		6,717,163	100.0

Table 5: **The twelve reasoning cells.** Point tasks score whether the predicted point falls in the target region; trajectory tasks score $\exp(-\mathrm{DFD})$, with DFD the discrete Fréchet distance between the predicted and reference 2D traces.

cell	benchmark	task	score
IA	IA-Bench (Blank et al., 2026)	point to the object a robot interacts with, or to a vacant goal location, in real robot frames (pointing version of the test split, 1,599 questions)	point in box
RR-A, RR-B	RoboRefIt (Lu et al., 2023)	referring-expression grounding in cluttered scenes, seen (testA) and unseen (testB) categories	point in target
RS-loc, RS-pl, RS-un	RefSpatial-Bench (Zhou et al., 2025a)	multi-step spatial referring: location, placement, unseen relations	point in target
VAB-P	VABench-P (Yuan et al., 2026a)	placement points for manipulation instructions	point in target
W2P	Where2Place (Yuan et al., 2024)	free-space placement points	point in mask
RSH	RoboSpatial-Home (Song et al., 2025)	point to a region that satisfies a spatial relation (context split)	point in mask
RI-T	RoboInter-VQA (Li et al., 2026)	object trajectory, contact and box prediction (validation split)	$\exp(-\mathrm{DFD})$
SR-T	ShareRobot-Bench (Ji et al., 2025)	end-effector trajectory	$\exp(-\mathrm{DFD})$
VAB-T	VABench-V (Yuan et al., 2026a)	8-point manipulation trajectory	$\exp(-\mathrm{DFD})$

Layer split. Making only layers 0–13 bidirectional and keeping 14–27 causal (same parameter count as the parent) reaches a training loss of 0.87 against 0.46 for the causal run at identical FLOPs and scores only 0.278 on the aggregate.

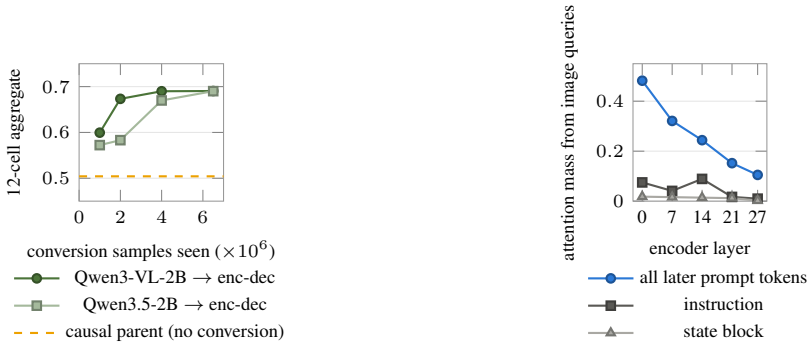


Figure 5: **Left:** 12-cell reasoning aggregate along the conversion run for Qwen3-VL-2B (the P6 backbone) and Qwen3.5-2B; the dashed line is the unconverted Qwen3-VL-2B parent. **Right:** fraction of image-query attention landing on later prompt tokens, by encoder layer of the converted encoder before policy training (24 DROID prompts). Under a causal mask this is zero by construction.

Table 6: **Embodied spatial reasoning, all reported backbones** (12-cell aggregate of Table 5). The layer split makes only layers 0–13 bidirectional. The Qwen3.5-2B conversion is a separate model without a matched causal twin; its aggregate (0.6901) is within rounding of the Qwen3-VL-2B encoder (0.6904) by coincidence.

Model	Size	Conv. data	12-cell Agg.
Baselines and controls			
Qwen3-VL-2B (causal parent; P1, P3)	2.1B	none	0.504
+ EFT mixture, causal (P2, P4)	2.1B	6.5M	0.657
+ EFT mixture, single-stack PrefixLM (P5)	2.1B	6.5M	0.660
Layer-split 14 (param-matched)	2.1B	6.5M	0.278
Qwen3-VL-4B + EFT mixture, causal	4.4B	6.5M	0.692
Converted encoders (ours)			
Qwen3-VL-2B converted (P6)	3.5B[a]	6.5M	**0.690**
Qwen3.5-2B converted	2B+2B[a]	6.5M	**0.690**

[a]Encoder-decoder during pretraining (both stacks share the vision tower and embedding table); the policy keeps only the encoder, 2.1B for Qwen3-VL-2B.

D PROBE DETAILS

Frames. The offline probes (attention, patching, representation change) run on a fixed set of 494 LIBERO-Plus frames recorded during evaluation rollouts, covering all four suites and seven perturbation types; each caption names the subset of 120 to 160 frames it uses. Probes of the backbone before policy training use 24 DROID prompts (Figure 5). Intervals are 95% bootstraps over the 40 LIBERO base tasks.

Modality-block control. Some attention analyses add a control outside the six paradigms: P1's weights trained with a block-causal attention mask over the agent view, the wrist view, the instruction and the state, so that each block sees itself and the blocks before it but never later ones. It asks whether a partial opening of the mask helps without pretraining; it does not, reaching 70.5% on LIBERO-Plus against 73.9% for P1 (Table 8).

Attention-path knockouts. We remove one attention path at test time by masking its query→key block before the softmax, which zeroes it and re-normalises the rest (the attention knockout of Geva et al. (2023)), and roll out the 4,000 knockout episodes of Appendix B. Because a knockout needs an explicit attention mask, we run each policy with a dense re-implementation of its attention. Without any knockout this changes the actions by about 1% and success by at most 1.2 points, far below the 10 to 48 points lost when an instruction path is removed, so the drops come from the removed path.

Further probes. *Representation change.* We swap the instruction for a different task while holding images and state fixed and report the relative RMS change of the final-layer image-token hidden states, $\Delta h = \|\mathbf{h}^{\text{img}}_{\text{swap}} - \mathbf{h}^{\text{img}}_{\text{orig}}\|_2 / \|\mathbf{h}^{\text{img}}_{\text{orig}}\|_2$; under a causal mask $\Delta h{=}0$ by construction. *Image-only head.* Under the same swap, the sensitivity ratio $R = \|\mathbf{a}^{\text{img}}_{\text{swap}} - \mathbf{a}^{\text{img}}_{\text{orig}}\|_2 / \|\mathbf{a}^{\text{full}}_{\text{swap}} - \mathbf{a}^{\text{full}}_{\text{orig}}\|_2$ compares the action change of a head restricted to image tokens with that of the full head; $R{\approx}1$ means the image tokens carry the full instruction effect. *Per-modality bottleneck.* We replace the backbone's full hidden sequence with 24 pooled tokens (8 per camera, 4 for language, 4 for state) produced by learned linear poolers trained jointly with the policy. If image tokens carry the instruction, bidirectional paradigms should tolerate this compression; causal paradigms, whose image tokens hold no instruction, should not.

Attention-map visualisation. For each model and map type, we identify positional sink cells whose location is the maximum in more than 30% of frames, remove those cells, renormalise the remaining map, and scale each map to its own 99th percentile. This exposes within-map spatial structure but does not preserve absolute mass across panels, so the maps (Figure 12, Figure 15, Figure 13) are qualitative only.

E PER-BENCHMARK RESULTS

This section breaks the ladder of Table 1 down by perturbation, task group and state input. Text values are means over the two training seeds.

The learned masks gain where objects move, not where the image changes. On LIBERO-Plus, conversion gains most when the robot's initial state is rearranged (+15.9 points over P1) and when the objects are laid out differently (+7.5), little under light, background or sensor noise, and nothing under camera shifts (−0.9; Table 7, Table 8). These perturbations change where the task objects are relative to the robot, not what they look like, so the gain fits the main-text reading that the learned masks help tie the instruction to the right objects rather than making the policy visually robust. PrefixLM gains more under reworded instructions (+5.2 against +1.8 for conversion), while the two flips lose most there (−8.8 for P3, −11.9 for P4): the policies whose instruction attention is least selective (Section 4.5) are the ones hurt most by unfamiliar wording. The modality-block control, which opens the mask only partly and without pretraining, falls 2.8 points below P1, most under rearranged initial states (−8.6); an imposed attention mask hurts even when it is only partly bidirectional.

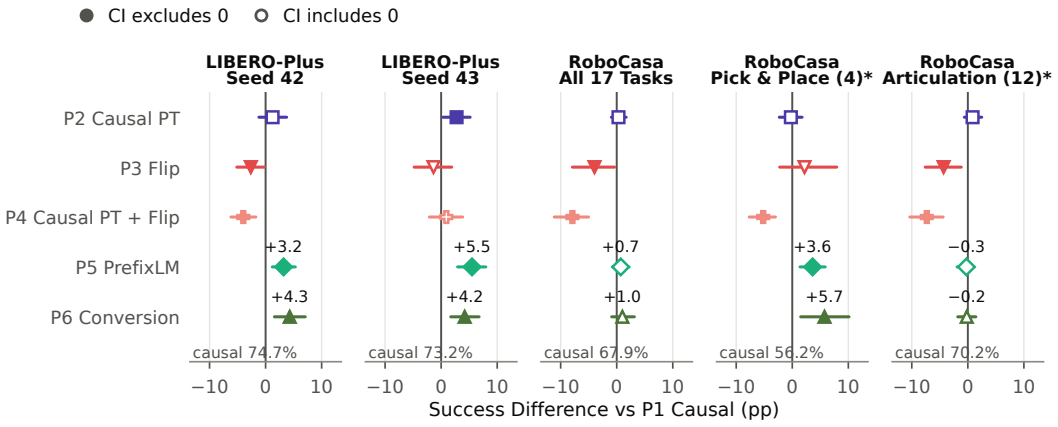


Figure 6: **Manipulation differences across LIBERO-Plus and RoboCasa365.** Paired difference to the causal base (P1), 95% interval over tasks. Filled markers: interval excludes zero. LIBERO-Plus: 10,030 episodes per seed. RoboCasa365: 17 tasks, two seeds, 200 clean episodes per task and seed; the pick-and-place and articulation split is post hoc.

Table 7: **Where each paradigm gains on LIBERO-Plus.** Paired difference to P1 of the same seed (points, seed 42 / seed 43), 10,030 instances, per suite (Goal) and per perturbation axis. **Bold**: the 95% interval over the 40 base tasks excludes zero at both seeds.

	All	Goal suite	Init. state	Layout	Language	Noise	Camera
P2	+1.2 / +2.8	+4.9 / +6.4	+4.1 / +7.4	+3.8 / +2.9	-0.3 / -0.4	+0.9 / +7.2	-1.6 / -3.0
P3	-2.6 / -1.4	-2.9 / +4.3	-4.2 / +0.8	+1.2 / -0.7	-11.5 / -6.0	-5.1 / -1.6	+2.9 / +1.1
P4	-4.0 / +0.9	-0.4 / +5.3	+1.2 / +5.8	+3.0 / +2.0	**-14.7 / -9.0**	-10.9 / +1.4	-3.9 / +6.9
P5	**+3.2 / +5.5**	**+8.8 / +10.8**	**+7.5 / +10.1**	+2.3 / +5.0	**+4.7 / +5.6**	+2.8 / +7.5	+0.8 / +2.9
P6	**+4.3 / +4.2**	**+6.8 / +10.3**	**+15.0 / +16.8**	**+8.3 / +6.7**	+2.5 / +1.1	+0.4 / +0.9	-0.4 / -1.3

The converted encoder uses the state more, and suffers more when it is wrong. Adding state tokens helps P6 and P1 about equally (3.6 and 3.1 points; Table 9), but replacing the state by random or zero values of the same format costs P6 16 to 21 points more than P1 (Table 10). Under a causal mask the state comes last and can only be read by the head; under a bidirectional one it also enters

Table 8: **P6 against P1 per perturbation axis on LIBERO-Plus** (success in %, Δ = P6 $-$ P1 in points). MB: the modality-block control (P1 weights with block-causal attention; Appendix G) minus P1, seed 42.

		seed 42			seed 43			MB
Perturbation	n	P1	P6	Δ	P1	P6	Δ	$-$P1 (s42)
Robot Initial States	1,550	50.5	65.5	$+15.0$	47.4	64.3	$+16.8$	-8.6
Objects Layout	1,525	75.1	83.3	$+8.3$	74.6	81.3	$+6.7$	-0.5
Language Instructions	1,537	82.5	85.0	$+2.5$	81.5	82.6	$+1.1$	-5.3
Light Conditions	1,142	91.9	94.1	$+2.3$	89.3	92.3	$+3.0$	-1.3
Background Textures	1,076	91.3	92.2	$+0.9$	92.1	93.4	$+1.3$	$+0.0$
Sensor Noise	1,601	82.3	82.6	$+0.4$	79.2	80.1	$+0.9$	-2.2
Camera Viewpoints	1,599	59.5	59.1	-0.4	58.4	57.1	-1.3	-0.5
All	10,030	74.7	79.0	$+4.3$	73.2	77.4	$+4.2$	-2.8

the image tokens (Figure 3b), so a wrong state corrupts the visual representation instead of being one input among several.

Table 9: **State tokens $\times$ attention mask on LIBERO-Plus** (P6 against P1 on the 4,000 knockout episodes of Appendix B, on which the state-off policies were evaluated). Mean $\pm$ standard deviation over two training seeds.

Condition	P6 (%)	P1 (%)	Δ (pp)
Without state	74.3 ± 1.7	71.0 ± 1.0	$+3.3 \pm 0.6$
With state	77.9 ± 0.7	74.1 ± 1.5	$+3.8 \pm 0.8$
State effect	$+3.6 \pm 2.4$	$+3.1 \pm 2.5$	interaction: $+0.5 \pm 0.2$

Table 10: **Format-preserving state corruptions on LIBERO-Plus** (P1 and P6, training seed 42). The same 1,000 instances (250 per suite) are rolled out clean and corrupted for both policies. Interaction: P6 drop $-$ P1 drop, 95% suite-stratified bootstrap.

	clean	shuffle	random	zero
P1 causal base	75.8%	71.0%	63.6%	64.7%
P6 conversion	79.2%	70.0%	50.7%	47.4%
interaction (pp)		$+4.4$ [$+0.6$, $+8.0$]	$+16.3$ [$+12.3$, $+20.6$]	$+20.7$ [$+16.7$, $+24.6$]

On RoboCasa365 the paradigms tie overall; the learned masks gain only where an object must be chosen. Averaged over 17 tasks the learned masks are within one point of P1 and the flips fall 4 to 8 points below it (Table 11). On the four pick-and-place tasks, where an object and a destination must be picked out among several, PrefixLM and conversion gain 3.6 and 5.7 points; on the twelve articulation tasks, where the scene fixes the target, no paradigm gains. The knockouts show that the same routes are at work as on LIBERO-Plus: without the forward route the causal policies lose nearly all of their success, and every bidirectional paradigm loses 41 to 50 points without the reverse route. PrefixLM depends least on the forward route (36.2 points lost), consistent with its small edit to the image tokens (Appendix G).

Without its instruction the converted encoder fails, the causal base does not. On plain LIBERO with an empty instruction, the causal base still solves 60% of Spatial and 58% of LIBERO-10 from the scene, whereas the converted encoder falls to 20% and 7% (Table 12). Both fail LIBERO-Goal without the instruction and never perform the scored task when given another Goal task's instruction, which is why the task-identity tests of RQ3 use LIBERO-Goal.

Table 11: **RoboCasa365 ladder** (17 tasks, 2 seeds, 200 clean episodes per task and seed). Δ is the paired difference to P1 averaged over tasks, with the post-hoc pick-and-place (P&P) and articulation groups of Appendix B separated. Knockout drops use 48 episodes per task and seed.

		success, Δ vs P1 (pp)				knockout drop (pp)	
		clean (%)	all 17	P&P (4)	articulation (12)	instr.→image	image→instr.
P1	causal base	67.9	0	0	0	67.3	–
P2	causal pretraining	68.2	$+0.3$	-0.3	$+0.8$	65.7	–
P3	flip	63.9	-4.0	$+2.2$	-4.3	60.8	46.6
P4	pretraining + flip	60.0	-7.9	-5.2	-7.3	57.0	41.1
P5	PrefixLM	68.6	$+0.7$	$+\mathbf{3.6}$	-0.3	**36.2**	49.6
P6	conversion	**68.9**	$+1.0$	$+\mathbf{5.7}$	-0.2	60.8	46.9

Table 12: **Instruction interventions on plain LIBERO** (success %, seed 42, 200 episodes per suite). No instruction: empty instruction string. Wrong Goal: each LIBERO-Goal task receives the next Goal task's instruction in its own scene.

	Spatial	Object	Goal	LIBERO-10
P1 causal, no instruction	60.0	42.5	8.0	58.0
P6 conversion, no instruction	19.5	45.5	9.5	7.0
P1 and P6, wrong Goal instruction			0.0	

F CLOSED-LOOP INSTRUCTION TRANSPLANT

The transplant asks whether a policy acts on the task written into its image tokens, not only whether the tokens carry it. It complements the patching and knockouts of the main text: patching shows that image tokens can transmit the instruction, knockouts that removing their route hurts, and the transplant that substituting them alone changes which task the robot performs.

Protocol. On LIBERO-Goal the ten tasks share one scene, so only the instruction identifies the task. At every policy step we run a second pass that differs only in its instruction (another Goal task, B), take its image-token states after layer 26, and substitute them into the policy's own pass; the instruction and state tokens keep their own values, and both passes share the flow-matching noise. In *override* the scene is task B's and the policy is told task A, so success means it performed the transplanted task; in *retention* the scene is task A's, so success means it kept its own task. Twenty episodes per task and paradigm (five for P1).

The scene does not give the task away. Running task B's scene with instruction A and no transplant yields 0 successes on B in 200 episodes for every bidirectional paradigm, and P1's transplant is exactly a no-op, since its image tokens cannot depend on the instruction (0 of 50). With the donor equal to the policy's own instruction the substitution leaves every action bitwise unchanged. Every override success therefore comes from the transplanted image tokens. Retention counts only whether task A still succeeds, not whether task B was attempted in A's scene.

The learned masks follow the transplanted task; the flips follow it less reliably. The learned masks perform the transplanted task in 78 to 93% of episodes and their own task in none (Table 13), so for them the image tokens decide the task. The flips perform it in 49 to 70% of episodes; their failures concentrate on two tasks (open the middle drawer, put the wine bottle on the rack) that P3 never performs under transplant, which suggests that the flips' image tokens encode some tasks only weakly. Repeating the transplant at the output of layer 27, the layer the head reads, leaves the learned masks unchanged (84 and 76%, still 0% retention) and lowers the flips further (55 and 45%, with 9.5 and 4.5% retention), so the flips' shortfall is not caused by the last layer re-reading their own instruction.

Table 13: **Transplant success per LIBERO-Goal task** (%, 20 episodes per task; P1: 5). B: success on the transplanted task (override); A: success on the policy's own task (retention). Last column: mean, 95% bootstrap over the ten tasks.

		task										
		0	1	2	3	4	5	6	7	8	9	mean
P1 causal	B	0	0	0	0	0	0	0	0	0	0	0 [0, 0]
	A	100	100	100	100	80	100	100	100	100	100	98 [94, 100]
P3 flip	B	0	95	80	70	65	85	100	100	100	0	70 [44, 90]
	A	0	30	0	0	0	0	0	0	0	0	3 [0, 9]
P4 PT + flip	B	0	85	55	5	85	10	55	95	85	15	49 [26, 70]
	A	10	0	0	0	10	0	0	5	0	0	2 [0, 5]
P5 PrefixLM	B	40	100	95	100	95	65	100	90	80	55	82 [68, 94]
	A	0	0	0	0	0	0	0	0	0	0	0 [0, 0]
P5, seed 43	B	75	95	100	100	95	90	95	100	100	80	93 [87, 98]
	A	0	0	0	0	0	0	0	0	0	0	0 [0, 0]
P6 conversion	B	75	60	90	85	90	85	75	100	80	40	78 [67, 87]
	A	0	0	0	0	0	0	0	0	0	0	0 [0, 0]
P6, seed 43	B	90	100	95	60	100	80	100	100	85	40	85 [72, 95]
	A	0	0	0	0	0	0	0	0	0	0	0 [0, 0]

G WHERE THE INSTRUCTION IS CARRIED AND HOW IT IS USED

This section adds the controls and breakdowns behind RQ2 to RQ4. Table 14 collects the per-paradigm measurements; the paragraphs below state what each additional probe adds. Text values are means over the two training seeds unless a seed is named.

Table 14: **What each policy relies on.** *Object lift*: instruction-to-image attention mass on the segmented task objects divided by their area share (1 = uniform; equation 3), clean state-matched LIBERO-Plus scenes of the 40 base tasks, separately for the agent view (camera 0) and the wrist view (camera 1). *Knockout*: success lost when the path is removed at test time, on the 4,000 knockout episodes of LIBERO-Plus (Appendix B), paired by episode, seed 42; at seed 43 the forward (instruction→image) knockout costs P1 27.7, P2 53.1 and P6 37.7 points (the other paradigms were not rerun at seed 43). *Patching*: share of an instruction swap's effect on the actions carried by all image tokens at layer 26 (one before the head input) (120 to 160 LIBERO-Plus frames per policy). *Image-only head*: share of the head's instruction sensitivity that survives when it reads image tokens only (seed 42). *Head attention on instruction*: fraction of the GR00T action head's attention mass that lands on the instruction tokens (240 or 494 LIBERO-Plus frames). Lift, patching and head attention: mean over two training seeds, with the two seed values in brackets. "constr.": zero by construction of the causal mask.

	object lift, instr.→image ↑		knockout drop (pp)		patching	image-only head	head attention
	agent view	wrist view	instr.→image	image→instr.	image share	instr. kept	on instruction
P1 causal base	1.74 [1.72, 1.77]	1.64 [1.43, 1.84]	31.9	0 (constr.)	0 (constr.)	0% (constr.)	0.13 [0.13, 0.14]
P2 causal pretraining	2.34 [2.31, 2.38]	1.51 [1.48, 1.53]	60.0	0 (constr.)	0 (constr.)	0% (constr.)	0.17 [0.16, 0.17]
P3 flip	1.33 [1.31, 1.36]	0.99 [0.95, 1.03]	9.9	45.8	0.86 [0.83, 0.88]	98%	0.03 [0.02, 0.04]
P4 pretraining + flip	1.57 [1.55, 1.60]	1.28 [1.22, 1.35]	66.6	40.7	0.69 [0.69, 0.70]	100%	0.02 [0.01, 0.03]
P5 PrefixLM	**2.76** [2.74, 2.78]	1.71 [1.68, 1.75]	23.8	47.4	0.67 [0.64, 0.70]	98%	0.15 [0.15, 0.15]
P6 conversion	**2.77** [2.64, 2.91]	1.58 [1.39, 1.76]	32.6	45.9	0.69 [0.66, 0.73]	94%	0.12 [0.09, 0.14]

The attention mask alone opens the image route only partly; training decides how it is used. Running the causal P1 and P2 under a bidirectional mask at probe time only, without any training under it, moves about half of an instruction swap's effect into the layer-26 image tokens (0.43 to 0.48), but almost none into the layer-12 tokens (0.06 to 0.11), where the trained flip P3 already carries 0.75 (Table 15). The untrained flip also changes the policy itself: it moves the predicted action about twice as far as an instruction swap does. The mask thus supplies a channel through which the instruction can reach the image tokens, while training decides how early and how much of it is used; the probe-time shares are an upper bound on what the mask alone provides.

Table 15: **Instruction-swap patching under an attention-mask flip at probe time only** (LIBERO-Plus frames, 160 per policy; seed 42 / seed 43). Image share: fraction of the swap's action effect transmitted by the image tokens. Effect: mean relative action change of the full instruction swap (native policy in brackets). Displacement: relative action change caused by the flip itself.

	image share L12	image share L26 (agent / wrist, s42)	effect	displacement	nearer own action
P1 flipped at probe time	0.07 / 0.06	0.46 (0.20 / 0.27) / 0.43	0.31 / 0.32 (0.25 / 0.27)	0.65 / 0.62	48 / 49%
P2 flipped at probe time	0.11 / 0.07	0.46 (0.20 / 0.29) / 0.48	0.45 / 0.59 (0.36 / 0.34)	0.64 / 0.69	48 / 41%
P3 (trained flip)	0.71 / 0.79	0.83 / 0.88	0.37 / 0.30		
P5, P6 (learned)	0.24–0.33	0.64–0.73	0.37–0.43		

The image route carries task identity: without it, LIBERO-Goal collapses. Splitting the knockouts by suite (Table 17) shows that removing the reverse path, from image tokens to the instruction, drops LIBERO-Goal to 6 to 22% in every bidirectional policy, from 68 to 79%. Goal is the suite in which only the instruction identifies the task, so this is the suite-level counterpart of the transplant (Appendix F). The reverse path also matters on the suites a scene largely determines, unlike in $\pi_{0.5}$, which loses mainly Goal. The reverse attention itself (Table 16) separates the two groups by amount, not by selectivity: in both, the agent-view object tokens read the instruction 1.2

to 1.65 times more than the rest of the image, but the flips' image tokens read it with about twice the mass (0.09 against 0.05 per token). The flips mix in more of the instruction, not more selectively.

Table 16: **Reverse-direction object lift** (image queries to instruction keys; clean LIBERO-Plus frames, 40 base tasks; seed 42 / seed 43, 95% CI over base tasks for the agent view). Mass: mean image-to-instruction attention per image token. Causal paradigms are 0 by construction. Forward lift: equation 3, agent view.

	agent-view reverse lift	agent-view mass	wrist reverse lift	wrist mass	agent-view forward lift
P1	0	0	0	0	1.72 / 1.77
P2	0	0	0	0	2.38 / 2.31
P3	1.36 [1.30, 1.44] / 1.28 [1.23, 1.33]	0.089 / 0.090	0.90 / 0.87	0.149 / 0.146	1.31 / 1.36
P4	1.51 [1.41, 1.60] / 1.65 [1.53, 1.76]	0.104 / 0.079	0.96 / 1.08	0.142 / 0.118	1.55 / 1.60
P5	1.39 [1.34, 1.46] / 1.21 [1.17, 1.27]	0.043 / 0.046	1.20 / 1.11	0.057 / 0.059	2.74 / 2.78
P6	1.31 [1.27, 1.35] / 1.43 [1.37, 1.50]	0.054 / 0.051	1.00 / 1.00	0.060 / 0.060	2.64 / 2.91

Table 17: **Knockouts per LIBERO suite** (success %, spatial / object / goal / long; LIBERO-Plus, seed 42, 1,000 paired episodes per suite). Reverse: image queries lose the instruction and state keys; forward: instruction and state queries lose the image keys. Goal excess: drop on Goal minus the mean drop on the other suites (points, 95% CI over the 10 base tasks per suite). $\pi_{0.5}$: plain LIBERO, 50 episodes per suite.

	clean	reverse removed	Goal excess (reverse)	forward removed	Goal excess (forward)
P1	82 / 82 / 70 / 67	(0 by mask)		46 / 62 / 24 / 42	$+19.6$ [$+7.7$, $+30.7$]
P2	81 / 83 / 75 / 64	(0 by mask)		7 / 16 / 14 / 2	-7.3 [-22.6, $+6.6$]
P3	81 / 78 / 68 / 62	42 / 47 / 6 / 18	$+23.3$ [$+13.2$, $+33.5$]	70 / 74 / 47 / 53	$+12.2$ [-0.6, $+25.2$]
P4	75 / 76 / 70 / 62	48 / 34 / 19 / 16	$+13.4$ [-3.8, $+28.7$]	0 / 2 / 10 / 0	-9.6 [-20.9, $+1.4$]
P5	84 / 82 / 79 / 67	35 / 41 / 10 / 24	$+25.2$ [$+12.9$, $+37.3$]	44 / 79 / 49 / 44	$+8.4$ [-6.2, $+24.2$]
P6	83 / 85 / 76 / 70	45 / 41 / 22 / 15	$+7.6$ [-8.4, $+22.9$]	59 / 55 / 45 / 18	-4.3 [-20.3, $+11.4$]
$\pi_{0.5}$	100 / 98 / 98 / 92	78 / 80 / 22 / 90		100 / 94 / 98 / 94	

The learned masks keep a second route through the instruction tokens; the flips give it up. The instruction can also reach the head through its own tokens. Patching the instruction tokens (Table 18) shows that this direct route carries most of the effect in the causal paradigms (0.49 to 0.81) and much less in the bidirectional ones. On RoboCasa365 the learned masks keep a substantial direct route (0.37 and 0.44), whereas the flips nearly lose it (0.07 and 0.13); on LIBERO-Plus both keep little (0.07 to 0.16). The flips' action heads, moreover, put only 0.02 to 0.03 of their attention on the instruction tokens, against 0.12 to 0.17 for every other paradigm (Table 14). The flips therefore depend on a single, less selective route, while the learned masks keep two; π_0 and $\pi_{0.5}$ likewise keep about 0.3 of the effect in their text tokens next to about 0.5 in their image tokens.

Table 18: **Share of an instruction swap's effect carried by the instruction tokens themselves** at layer 26 (activation patching of the instruction-token group). The group can be patched only on frames whose two instructions have the same token count: 11 of 120 LIBERO-Plus frames for P2 to P5 and 19 of 120 RoboCasa365 frames; P1 and P6 on LIBERO-Plus use the matching frames of a 160-frame bank. Seed values: LIBERO-Plus seeds 42 / 43, RoboCasa365 seeds 4 / 42.

	LIBERO-Plus (s42 / s43)	RoboCasa365 (s4 / s42)
P1 causal base	0.72 / 0.80	0.85 / 0.76
P2 causal pretraining	0.51 / 0.46	0.48 / 0.60
P3 flip	0.09 / 0.04	0.21 / 0.04
P4 pretraining + flip	0.13 / 0.10	0.09 / 0.05
P5 PrefixLM	0.19 / 0.13	0.44 / 0.30
P6 conversion	0.11 / 0.20	0.40 / 0.47

How much a policy leans on each route depends on its training. Figure 7 places every policy by the success lost without each route. Causal pretraining (P2) moves onto the forward route alone, every bidirectional paradigm also needs the reverse route, and π_0 and $\pi_{0.5}$ need only the reverse one (0.5 and 3.0 points lost without the forward route, 29.5 and 59.2 without the reverse). The style of the image-token edit also depends on training (Figure 8): PrefixLM and $\pi_{0.5}$ change their image tokens little under an instruction swap (0.23 and 0.26 relative change) yet carry the full effect, and PrefixLM survives the forward knockout on RoboCasa365 far better than conversion (36.2 against 60.8 points lost). On RoboCasa365 the dependence also varies by task (Figure 9). Compressing the backbone output to 24 pooled tokens costs the causal paradigms at most 1.8 points and sharpens the converted encoder's agent-view lift from 2.77 to 3.19, consistent with image tokens that already carry the instruction.

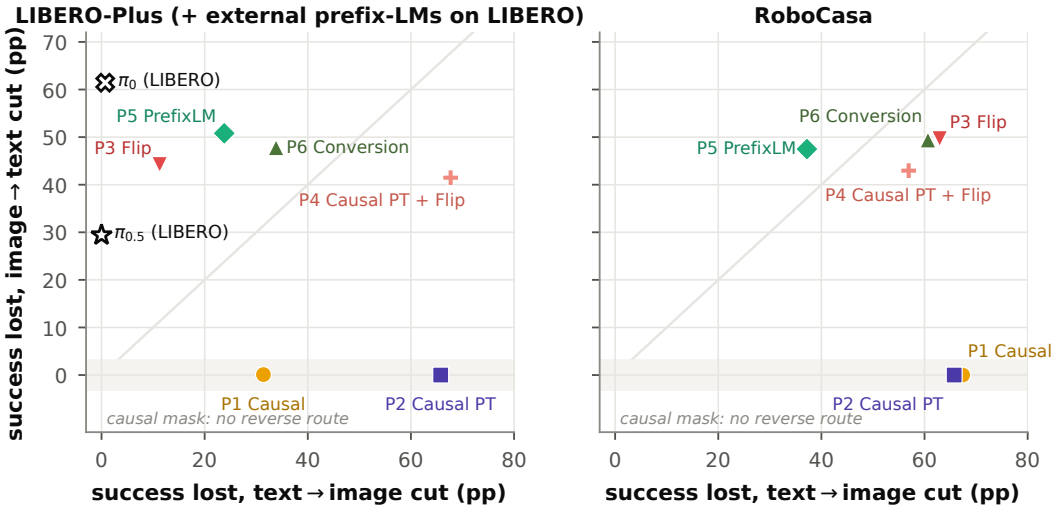


Figure 7: **Where the cross-modal routes matter.** Success lost when text tokens can no longer read the image (x-axis) against success lost when image tokens can no longer read the text (y-axis). Causal paradigms lie on the x-axis by construction. Causal pretraining (P2) moves far right, onto a single text-to-image route. Every bidirectional paradigm also relies on the reverse route, and the independently trained π_0 and $\pi_{0.5}$ rely on it alone. LIBERO-Plus: seed 42, 4,000 episodes per cell. RoboCasa365: two seeds, 17 tasks. π: LIBERO, 5 trials per task.

Pretraining makes cross-modal attention sharper and more consequential, not larger. The flip without pretraining, P3, sends more image attention to the instruction than the converted P6 (summed mass 4.19 against 2.40; Table 19) and keeps reading it up to the last layer, whereas P6's cross-modal attention concentrates in layers 0 to 18 (Figure 11). Yet P6 depends far more on the forward route (32.6 against 9.9 points lost; Table 20) and its actions respond more to an instruction swap (0.43 against 0.36). Already before policy training, the converted encoder sends as much mass to the instruction as an unconverted bidirectional copy of the parent but distributes it more sharply (normalised entropy 0.65 against 0.79). Mixing more is thus not the same as using the instruction more; what pretraining under the mask adds is where the attention goes.

Patching by token group. Figure 10 shows the per-group patching shares and the lift on both cameras. On the wrist view, where the task objects fill a larger part of the image, lifts are compressed towards 1 and the paradigms other than the flips overlap (1.51 to 1.71), while the flips stay lowest on both cameras. The patching shares are projections, and token groups overlap in their downstream effects, so they need not sum to one.

Open-loop stability does not explain the gain. Both bidirectional policies keep their action-head attention more stable under sensor noise than P1 (map correlation 0.90 for P3 and 0.84 for P6, against 0.78; Figure 12) and change their action chunks less under task-preserving perturbations while still following a changed instruction. Because P3 matches P6 on these measures yet loses 6.0

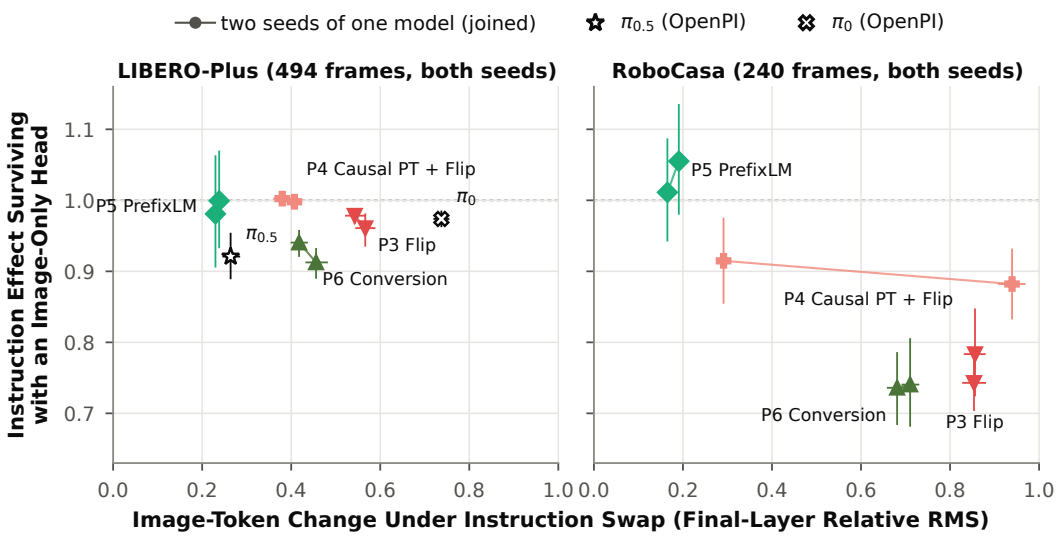


Figure 8: **How the image tokens carry the instruction depends on training.** Change of the final image tokens under an instruction swap (x-axis) against the share of the head's instruction effect that survives an image-only head (y-axis). Both seeds shown and joined. PrefixLM and $\pi_{0.5}$ make a small edit that carries the full effect. The converted encoder, the mask flip, and π_0 rewrite the image tokens more heavily.

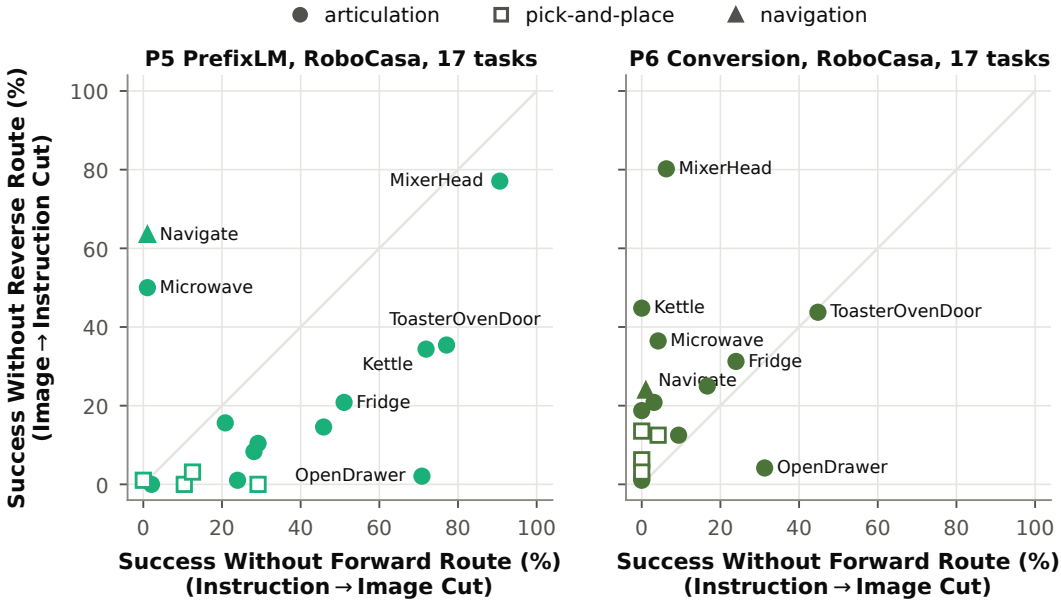


Figure 9: **Each RoboCasa365 task runs mainly through one route.** Per-task success without the forward route (x-axis) and without the reverse route (y-axis), mean of two seeds, 48 episodes per task and seed. PrefixLM keeps single-object articulation tasks without the forward route but loses navigation and the microwave without the reverse route. Pick-and-place tasks (squares) need both routes. Differences under about 15 pp are within noise.

points of closed-loop success, stability under perturbation is a property of the mask, not the source of the gain.

Table 19: **Image-token attention and swap sensitivity on LIBERO-Plus frames.** Attention mass from image queries summed over the 28 backbone layers; entropy of image→instruction attention (1 = uniform), mean over layers; Δh: relative RMS change of final-layer image tokens when the instruction or state is swapped (images fixed); Δa: relative L2 change of the predicted action chunk. Seed 42. MB: modality-block control (P1 weights, block-causal attention). The first two columns and the Δh columns are 0 by construction for P1 and MB.

Policy	img→instr.	img→state	img→other cam.	entropy	Δh instr. (%)	Δh state (%)	Δa instr.	Δa state
P1 causal base	0.00	0.00	1.85	–	0.0	0.0	0.25	0.34
MB (P1 weights)	0.00	0.00	1.99	–	0.0	0.0	0.25	0.34
P3 flip	4.19	2.16	5.05	0.72	54.3	59.3	0.36	0.41
P6 conversion	2.40	1.41	2.90	0.71	41.8	41.2	0.43	0.48

Table 20: **Attention-path knockouts** for P1, P3 and P6 (LIBERO-Plus, 4,000 knockout episodes, seed 42, paired by episode; 95% CI over the 40 base tasks). The forward and reverse drops of P2, P4 and P5 are in Table 14.

Paradigm	knocked-out path (query→key)	clean	knockout	drop (pp)	95% CI
P1	image→instr.+state	75.15	75.05	+0.10	[−1.07, +1.26]
P1	instr.→image	75.15	43.30	+31.85	[+25.71, +37.98]
P1	instr.+state→image	75.15	43.75	+31.40	[+25.11, +37.64]
P3	image→instr.	72.42	26.62	+45.80	[+38.82, +52.56]
P3	image→state	72.42	70.88	+1.55	[−0.37, +3.54]
P3	image→instr.+state	72.42	28.18	+44.25	[+37.13, +51.21]
P3	instr.→image	72.42	62.55	+9.88	[+5.29, +15.11]
P3	state→image	72.42	71.70	+0.72	[−0.34, +1.79]
P3	instr.+state→image	72.42	61.15	+11.28	[+6.26, +16.99]
P6	image→instr.	78.40	32.52	+45.88	[+38.36, +53.32]
P6	image→state	78.40	76.25	+2.15	[−0.02, +4.73]
P6	image→instr.+state	78.40	30.62	+47.77	[+40.29, +55.21]
P6	instr.→image	78.40	45.83	+32.58	[+24.67, +40.86]
P6	state→image	78.40	78.20	+0.20	[−0.80, +1.14]
P6	instr.+state→image	78.40	44.55	+33.85	[+26.01, +42.06]
P3	none (dense probe)	72.42	71.28	+1.15	[+0.02, +2.30]
P6	none (dense probe)	78.40	78.53	−0.12	[−1.06, +0.81]

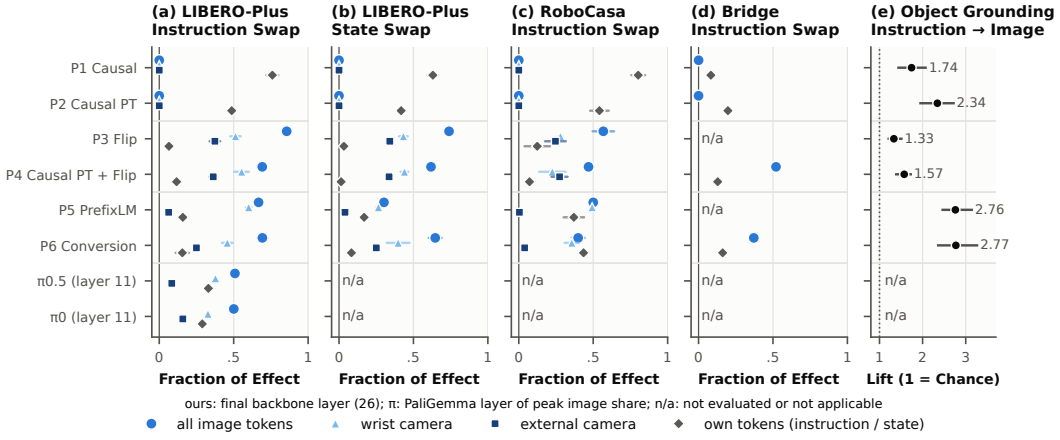


Figure 10: **Which token groups carry an input swap, and whether attention lands on task objects.** Panels (a–d) show activation-patching fractions at layer 26; bars span the two seed means where available. Panel (e) reports instruction-attention lift on segmented task objects, with bars spanning the per-seed 95% intervals. External π models are shown at their peak image-carrier layer because their action expert reads every prefix layer.

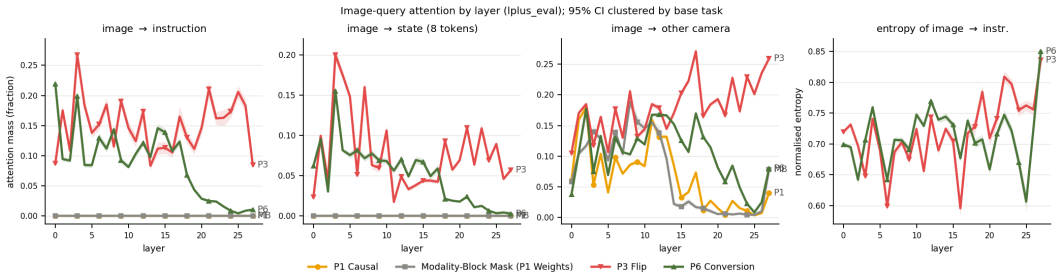


Figure 11: Image-query attention by backbone layer on LIBERO-Plus frames (seed 42; 95% CI over base tasks). P1 and the modality-block control (MB: P1 weights, block-causal attention) are zero in the first two panels by construction; under the modality-block mask the agent view cannot attend to the wrist view.

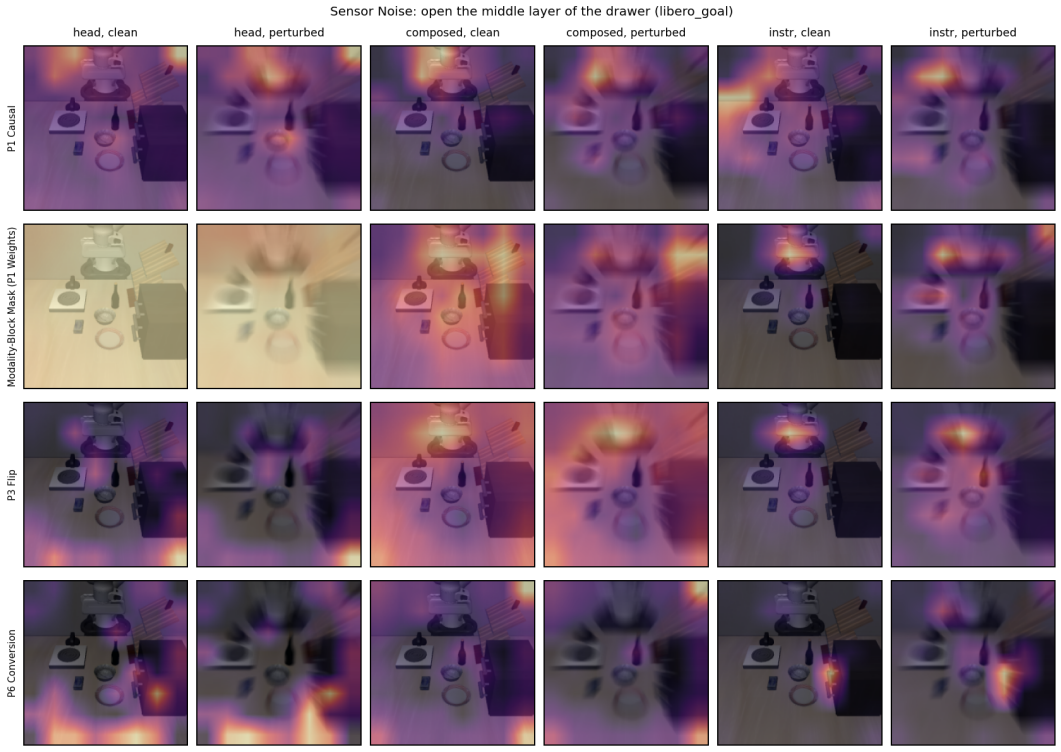


Figure 12: Attention maps on one LIBERO-Plus state, clean vs sensor noise (seed 42; rows P1, modality-block control, P3, P6). Head: action-query attention of the GR00T head; composed: head attention propagated through the last backbone layer; instr: instruction-query attention. Instruction (BDDL wording): "open the middle layer of the drawer". On this state only P6's instruction attention stays on the drawer handles; this is a single qualitative example.

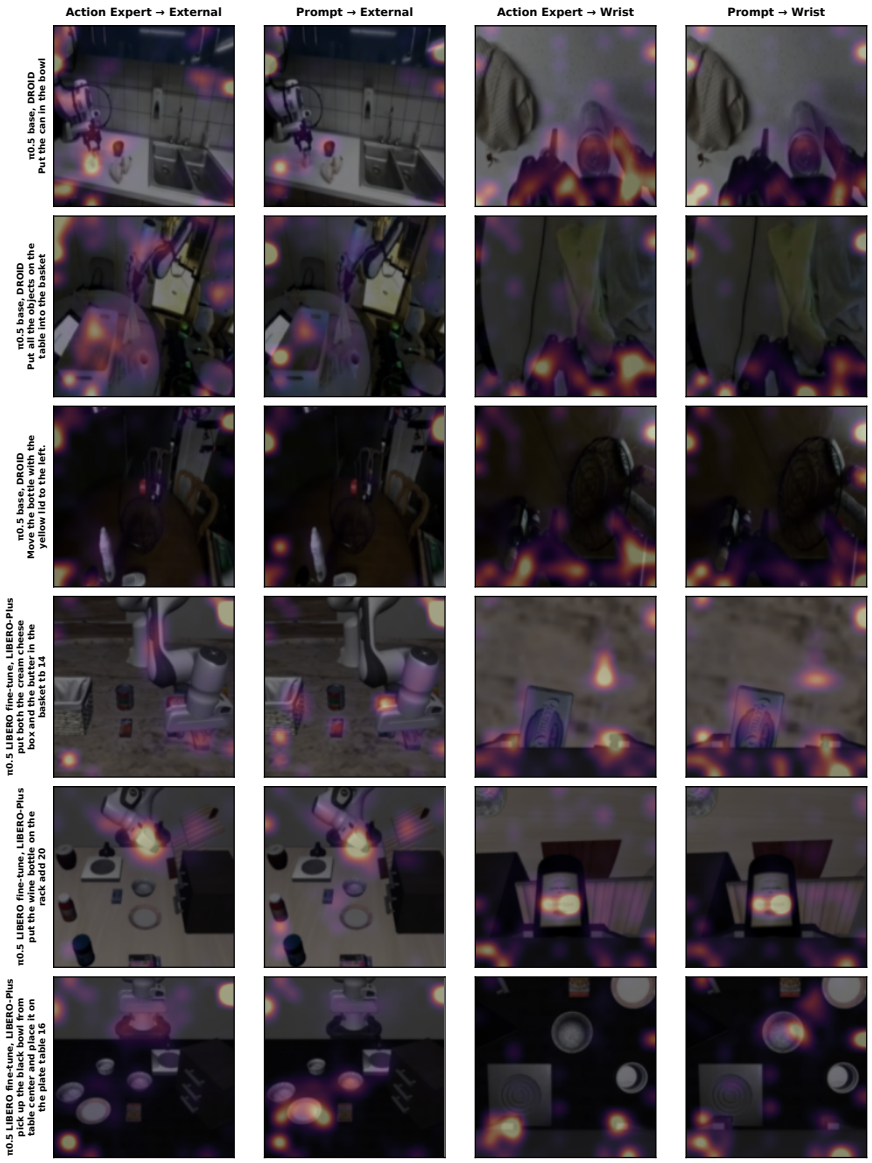


Figure 13: $\pi_{0.5}$ **attention maps before and after LIBERO fine-tuning.** Top rows use real DROID frames with the base model; bottom rows use LIBERO-Plus frames with the fine-tuned policy. Columns show action-expert and prompt attention to the external and wrist cameras. Positional sinks are removed and each map is independently scaled.

H BRIDGE AND SIMPLERENV

Bridge tests whether the simulation results carry over to policies trained on real images with a single third-person camera. With one training seed and three to four tasks per scene set, it is the weakest evidence in the paper, and we read it for direction rather than size.

Protocol. All six paradigms train on BridgeData V2 with the recipe of Appendix A. Stock scenes are the SimplerEnv WidowX visual-matching setup (four tasks, 24 scenes, five rounds; 480 episodes). For the three table tasks we also generate 96 fixed new scenes per task, once with a random layout and once with two added distractor objects taken from the other tasks, identical for every policy. In the eggplant scene the robot starts above the height range of the training data, so for that task every state-reading policy receives its end-effector height shifted by -0.093 m.

Conversion's lead over causal pretraining comes from telling objects apart. Conversion and causal pretraining on the same data are close on the stock scenes ($+2.5$ points, $z = 0.8$) and separate once distractors are added ($+7.6$, $z = 2.3$; Table 21). Almost all of this comes from the stacking task, in which the instruction names one of two blocks by colour: conversion leads there by 18.8 points among distractors ($z = 3.1$), while on the carrot and spoon tasks the two are within 4 points (Table 22). This is the Bridge counterpart of the simulation pattern: the learned mask helps where the instruction must single out an object among similar ones.

Clutter pushes the instruction into the image tokens, but how much does not predict success. Distractors raise the image tokens' share of the instruction effect by about 0.06 in both bidirectional policies (P4 and P6; intervals over episodes exclude zero), yet within one policy the share is the same in successful and failed episodes (Figure 14). As in simulation, what matters is how selectively the image tokens use the instruction, not how much of it they carry: P4 carries more than P6 (0.55 against 0.40 among distractors) and succeeds less, and P3, whose image tokens carry the most (0.80), succeeds least (38.9% among distractors; Table 23).

PrefixLM does not transfer to a single third-person camera. PrefixLM, which ties conversion in simulation, stays near the causal base on Bridge (46.2% on the stock scenes, 15.6 points below conversion, $z = 4.0$) and falls below causal pretraining. Patching shows why its route is weak here: it carries only 0.10 of the instruction effect in its image tokens, against 0.34 for conversion, and keeps most of it in the template tokens, like the causal policies (Table 23). A plausible cause, which we have not tested, is the camera: on LIBERO-Plus PrefixLM's image route runs almost entirely through the wrist view (agent-view share 0.06 against 0.25 for conversion), and Bridge has no wrist camera. The external-view share on LIBERO-Plus orders P5, P6 and P4 exactly as the one-camera share on Bridge does.

The converted encoder relies on the state. Without state tokens, P6 loses 19.4 points on the stock scenes ($z = 4.5$) and every other paradigm at most 8.6, in line with its larger sensitivity to corrupted state on LIBERO-Plus (Appendix E).

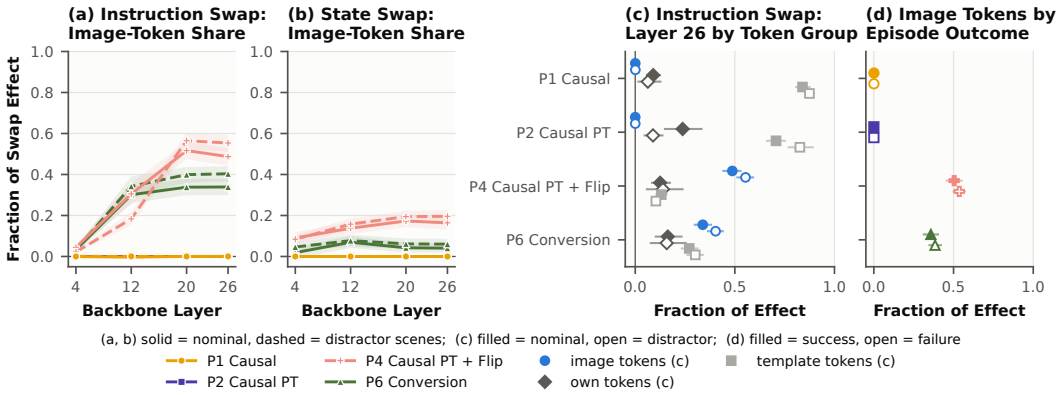


Figure 14: **Bridge activation patching on SimplerEnv rollouts.** Solid and filled marks denote stock scenes; dashed and open marks denote fixed scenes with new layouts and two distractor objects. Distractors increase the final image-token share by 0.064 $[0.009, 0.119]$ for P6 and 0.066 $[0.005, 0.124]$ for P4, but within one policy the share does not distinguish successful from failed episodes. Intervals bootstrap episodes.

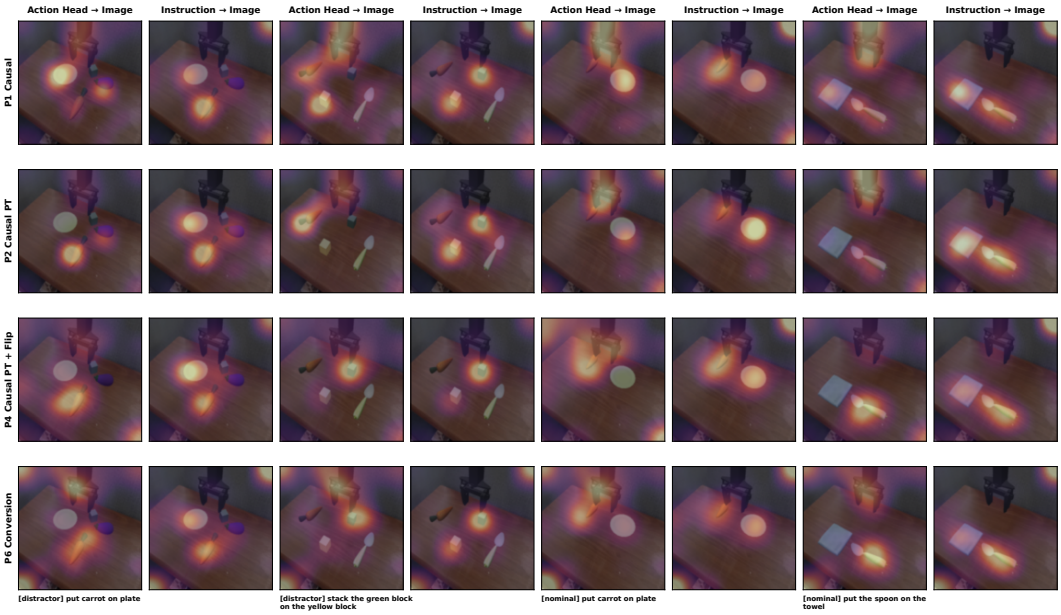


Figure 15: **Full SimplerEnv attention-map grid.** Rows are four of the Bridge paradigms (P3's and P5's maps exist but are not shown); columns pair action-head and instruction attention for two distractor and two stock scenes. Positional sinks are removed and each map is independently scaled, so only spatial structure within a panel is interpretable.

Table 21: **Bridge in SimplerEnv WidowX** (seed 42, with state, 30k steps; success in %). Stock: visual-matching scenes, 4 tasks × 24 scenes × 5 rounds (480 episodes). Distractor: 96 fixed new scenes per table task, each with two objects from the other tasks, identical for every policy (288 episodes). Stack: stock / distractor success on the colour-discriminated stacking task. Image share: final-layer patching share on 240 stock-scene and 240 distractor-scene frames, the same for every policy. Differences are paired over shared scenes, with z from the episode-level standard error.

	stock (%)	distractor (%)	stack: stock / dist. (%)	image share: stock / dist.
P1 causal base	45.0	45.8	4 / 15	0 / 0
P2 causal pretraining	59.4	49.3	13 / 14	0 / 0
P3 flip	42.1	38.9	19 / 26	0.82 / 0.80
P4 pretraining + flip	53.1	43.8	8 / 12	0.49 / 0.55
P5 PrefixLM	46.2	45.1	18 / 28	0.10 / 0.12
P6 conversion	**61.9**	**56.9**	**28 / 32**	0.34 / 0.40
P6 − P2 (pp, z)	+2.5 (0.8)	+**7.6** (2.3)		
P6 − P1 (pp, z)	+16.9 (4.5)	+11.1 (3.5)		
P6 − P3 (pp, z)	+19.8 (4.6)	+18.1 (4.9)		
P6 − P4 (pp, z)	+8.8 (2.3)	+13.2 (3.9)		
P6 − P5 (pp, z)	+15.6 (4.0)	+11.8 (3.4)		
P5 − P2 (pp, z)	−13.1 (3.7)	−4.2 (1.2)		

Table 22: **Bridge success per task** (%, seed 42, with state, 30k steps). Stock: 120 episodes per task (24 scenes × 5 rounds); random layout and distractor: 96 fixed scenes per task, identical for every policy. The eggplant task has no distractor variant. Last row: P6−P2 on the stacking task, paired over scenes, with z from the episode-level standard error.

	stock				random layout				distractor		
	carrot	spoon	stack	eggplant	carrot	spoon	stack	eggplant	carrot	spoon	stack
P1 causal base	42	64	4	69	50	59	11	72	48	75	15
P2 causal pretraining	54	72	13	98	49	66	28	98	55	79	14
P3 flip	27	47	19	76	43	40	17	83	33	57	26
P4 pretraining + flip	38	72	8	93	35	80	5	99	47	72	12
P5 PrefixLM	38	49	18	79	34	56	21	96	47	60	28
P6 conversion	44	83	**28**	92	47	74	**33**	98	55	83	**32**
stack: P6 − P2 (pp, z)			+14.2 (1.9)				+5.2 (0.8)				+18.8 (3.1)

Table 23: **Instruction-swap patching of the Bridge policies** (last layer, 240 stock / 240 distractor SimplerEnv frames). Shares of the swap's action effect carried by each token group; state: image share under a state swap.

	image tokens	own instruction tokens	template tokens	state swap: image
P1 causal	0 / 0	0.09 / 0.06	0.84 / 0.87	0
P2 causal PT	0 / 0	0.24 / 0.09	0.71 / 0.83	0
P3 flip	0.82 / 0.80	0.04 / 0.05	0.08 / 0.07	0.20 / 0.20
P4 PT + flip	0.49 / 0.55	0.12 / 0.14	0.13 / 0.10	0.16 / 0.20
P5 PrefixLM	0.10 / 0.12	0.17 / 0.20	0.55 / 0.64	0.10 / 0.11
P6 conversion	0.34 / 0.40	0.16 / 0.16	0.27 / 0.30	0.04 / 0.06

I TRANSFER ACROSS ACTION HEADS

All main-text policies use a GR00T-style action head that reads only the final backbone layer. This section asks whether the converted encoder's gain survives a π-style layer-wise head, PI_{v4} Community (2026), whose blocks each read a different backbone layer. Each pair compares P6 with P1 under otherwise identical training (LIBERO-Plus: 30k steps, state tokens normalised to the 1st and 99th percentiles for both; RoboCasa365: 50k steps, two seeds).

Table 24: **Encoder gain under two action heads.** Success (%) of the causal policy and the converted encoder (P6), and the encoder's gain in points, each $\pm$ its standard error over episodes. LIBERO-Plus: P1 and P6 with percentile-normalised state tokens, one row per training seed (the PI_{v4} seed-43 causal evaluation covers 8,258 of 10,030 episodes; that row uses only episodes both policies ran). RoboCasa365: causal = P1, two seeds pooled; P&P: the four-task post-hoc pick-and-place group; GR00T cells use 200 episodes per task and seed, PI_{v4} cells 48.

head	benchmark		causal	encoder	encoder gain
GR00T-style (final layer)	LIBERO-Plus	s42	76.1 ± 0.4	79.1 ± 0.4	$\mathbf{+3.0 \pm 0.6}$
	RoboCasa365	all 17	67.9 ± 0.6	68.9 ± 0.6	$+1.0 \pm 0.8$
		P&P	56.2 ± 1.2	61.9 ± 1.2	$\mathbf{+5.7 \pm 1.7}$
PI_{v4} (every layer)	LIBERO-Plus	s42	78.1 ± 0.4	78.9 ± 0.4	$+0.8 \pm 0.6$
		s43	75.6 ± 0.5	74.8 ± 0.5	-0.7 ± 0.7
	RoboCasa365	all 17	66.9 ± 1.2	65.9 ± 1.2	-1.0 ± 1.7
		P&P	62.8 ± 2.5	57.3 ± 2.5	-5.5 ± 3.5

With a layer-wise head the gain disappears, because the causal policy catches up. Switching from the GR00T-style head to PI_{v4} raises the causal policy on LIBERO-Plus (seed 42, where both heads were trained) from 76.1 to 78.1% while the encoder stays at about 79% (Table 24); pooled over two seeds the encoder's gain with PI_{v4} is $+0.1$ points $[-1.5, +1.8]$. On the RoboCasa365 pick-and-place tasks the pattern is the same (causal 56.2 to 62.8%, encoder 61.9 to 57.3%). The drop in the encoder's gain between the two heads ($+2.1$ points on LIBERO-Plus, $+11.2$ on pick-and-place) is suggestive rather than decisive: its task-level interval on LIBERO-Plus includes zero ($[-0.6, +4.8]$), and the pick-and-place group is small and post hoc.

Interpretation: the policy needs image-instruction fusion once. A bidirectional backbone performs the fusion in the image tokens that a final-layer head reads. A layer-wise head can perform it itself, since its action tokens attend to image and instruction keys at every depth, so a causal backbone paired with it loses little and the two sources of fusion do not add. Patching the PI_{v4} pair supports this reading: the encoder still carries the instruction in its image tokens (about a third of the effect at layer 12 and almost half at layer 20, mostly in the wrist view), but the causal policy compensates by relaying the instruction into the tokens that follow it. This also suggests why π_0 and $\pi_{0.5}$ work with a bidirectional prefix and a layer-wise expert, although no causal twin of either exists to test it. The heads differ in capacity as well as topology, so the reading remains a hypothesis, and our claims are restricted to heads that read the final backbone layer.

The causal layer-wise policy depends on the state suffix. On RoboCasa365, removing the state suffix (the state tokens and their markers after the instruction) collapses the causal PI_{v4} policy to 0.2% while the converted encoder keeps 58.4% (Table 25). This fits the LIBERO-Plus patching above, in which the causal policy relays the instruction into the tokens that follow it, and it is not an attention-allocation effect: both heads spend the same mass on the `[STATE]`/`[ACTION]` markers (0.024 each), the encoder head reads the state numbers more, not less (0.30 vs 0.25 per block), and removing the suffix moves roughly 0.16 of head mass onto image tokens in both backbones with an almost unchanged spatial distribution (per-camera map correlation 0.96 to 0.98), consistently across both training seeds.

Table 25: **State-suffix removal on RoboCasa365** (P1 and P6 with the PI_{v4} head, 17 tasks $\times$ 48 episodes, 2 seeds). Mean $\pm$ seed spread.

Backbone	Clean (%)	Suffix removed (%)	Drop (pp)
P1 causal base	66.9 ± 3.0	0.2 ± 0.3	-66.7
P6 conversion	65.9 ± 0.6	58.4 ± 6.3	-7.5

J EXTERNAL REFERENCE POINTS

Our aim is a controlled comparison of attention masks on one backbone, not a new state of the art. For orientation, Table 26 places our best policies next to published systems. The external numbers are as reported by their authors and differ from ours in model size, training data, horizon and recipe. Our best LIBERO-Plus policies, with a 2B backbone and no LIBERO-Plus-style augmentation, sit 5 to 7 points below the strongest system.

Table 26: **External reference points** (success %, as reported; not protocol-matched). LIBERO-Plus: zero-shot from LIBERO training unless marked. Ours: mean of two seeds. The RoboCasa365 rows use the official 50-task leaderboard (2026-09-12), which our 17-task atomic protocol does not match.

model	note	LIBERO-Plus	LIBERO
OpenVLA-OFT (Kim et al., 2025; Fei et al., 2026)	per-suite checkpoints	69.6	97.1
StarVLA Qwen3-VL-PI 4B (Community, 2026)	repository README	77.0	–
VLANeXt (Wu et al., 2026)		83.9	97.4
InternVLA-A1.5 (Ma et al., 2026)		84.8	98.9
OpenVLA-OFT+ (Fei et al., 2026)	LIBERO-Plus-style augmentation	79.6	–
SRPO (Senyu et al., 2026)	augmentation and RL	82.1	–
P5 PrefixLM (ours)	2B backbone	78.3	–
P6 conversion (ours)	2B backbone	78.2	–
		RoboCasa365 (50 tasks)	
π_0	leaderboard	14.8	
GR00T N1.5	leaderboard	23.9	
Xiaomi-Robotics-1	leaderboard	57.4	