DOES THE POLICY KNOW WHERE THE BOX IS?

Anonymous authors
Paper under double-blind review

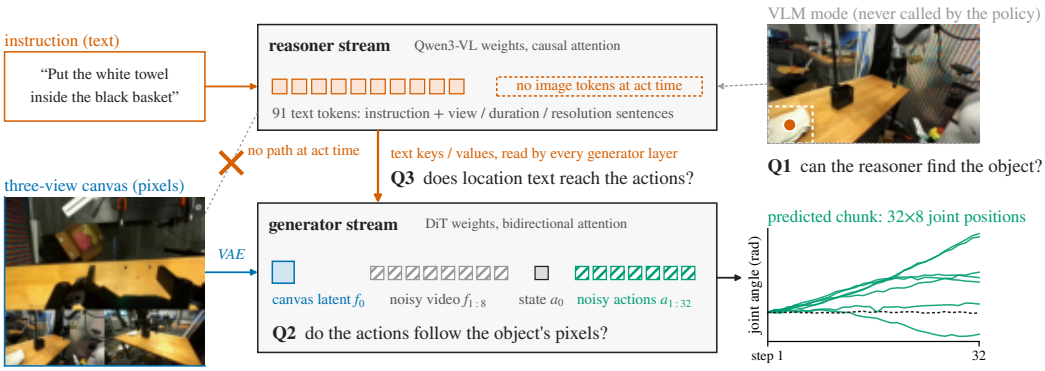


Figure 1: **The released Cosmos3-Nano-Policy-DROID at act time.** The reasoner stream receives only the 91-token instruction; the camera canvas reaches it only in VLM mode, which the policy never calls (top right: served there, the policy checkpoint's own reasoner does point at the target; dot: its point, dashed box: ground truth). The generator stream receives the canvas as a VAE latent, reads the reasoner only through its text keys and values, and emits a 32×8 joint chunk. Q1–Q3 mark the three questions we test; all inputs are one real D-DROID sample.

ABSTRACT

Unified world-action models place a vision-language reasoner and a video-action diffusion generator in one network, on the assumption that the policy inherits the reasoner's ability to ground objects. We test this assumption without training on the released Cosmos3-Nano-Policy-DROID, a 16B Mixture-of-Transformers built on a Qwen3-VL reasoner. Its inference code shows why the assumption could fail: at act time the reasoner stream receives only the text prompt, while the camera canvas enters only the generator, as VAE latents. Paired pixel, text, and activation interventions on 500 post-training frames (200 for activations) drawn from 2,302, and 500 paired closed-loop episodes per condition, show three things. Served in VLM mode, the policy checkpoint's own reasoner localizes the target on 71–76% of single views but 46% of the policy's three-view canvas (its base VLM: 76–78% and 50%); the halved resolution costs 0.14 in hit rate and the three-view layout 0.12. The named target is exposed far more sharply to a linear probe in the reasoner's image tokens than in any of the generator's token groups (top-cell hit 0.55 against at most 0.39; position-only prior 0.24), although both towers carry it (token AUROC 0.85 against 0.81; equal probe mass on the mask), yet the generator's actions follow the object's pixels and re-target when it is moved. The action head reads the instruction's noun: deleting or swapping it costs 39–46 percentage points (pp) of closed-loop success. It does not act on location text: a location sentence shifts actions about as much as a location-free one (0.5 cm more, toward neither location), survives a matched causal knock-out, steers toward neither instance on 26 frames with a verified same-class distractor (75 candidates), and gives no closed-loop gain (-0.4 pp, 95% CI $[-4.4, +3.4]$). The tower that exposes the object is not the tower that acts, and location text does not bridge them.

1 INTRODUCTION

Robot policies fail in specific, costly ways: told to fetch the mug, they knock over the cup beside it. Coupling a vision-language reasoner to the policy is meant to fix this, on the premise that a

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.

model which can already point to the right object will pass that ability to the actions it produces. The premise is seldom checked on a released model: task success cannot tell a policy that uses the reasoner's grounding from one that succeeds on other cues, and studies that look inside a VLA (Appendix A) ask what the backbone represents, not whether a capability crosses the interface at inference time.

A recent family of robot foundation models couples a pretrained vision-language model (VLM) with a generative action or video model inside one network (Black et al., 2025b; NVIDIA et al., 2025; Cheang et al., 2024; Li et al., 2025; NVIDIA, 2026); π_0, for one, motivates its pretrained VLM backbone as a way to inherit Internet-scale semantic knowledge (Black et al., 2025b). Cosmos 3 (NVIDIA, 2026) takes the design to its limit: one Mixture-of-Transformers (Liang et al., 2025) backbone holds a Qwen3-VL reasoner (Bai et al., 2025) and a diffusion generator, so that the reasoner's understanding is meant to be available to the generator, and the post-trained Cosmos3-Nano-Policy-DROID was reported as the best policy on the RoboArena leaderboard (Atreya et al., 2025) at release; whether it uses the reasoner's grounding has not been measured.

We ask this of the released checkpoint, without training, with interventions rather than correlations. The starting point is a fact about the inference code (Section 2.1): at act time the reasoner stream of Cosmos3-Nano-Policy-DROID receives only the text instruction plus fixed meta sentences, never image tokens, while the three-view canvas enters only the generator stream, as clean VAE latents beside noisy video and action tokens (Figure 1). "Does the policy know where the box is" therefore splits into three testable parts:

Q1 Can the reasoner localize the target on the frames the policy acts on, when it is allowed to see them?

Q2 Does the generator encode the target's location, and do predicted actions causally follow the object's pixels?

Q3 If the reasoner's localization is handed to the action head through the only channel it has, the prompt, is it used?

We build D-DROID, 2,302 frames drawn from the exact data Cosmos3-Nano-Policy-DROID was post-trained on, with detector-and-segmenter ground truth, stored per-sample noise seeds, and paired interventions in pixels, text, and activations (Figure 2b; Figure 3 in Appendix F). Our contributions are:

- A causal protocol for grounding in unified world-action models: paired pixel edits with artifact-matched controls, paired text edits with a location-free control, and fixed noise seeds isolate object-following from generic input sensitivity (Section 2).
- The reasoner grounds well on single views but poorly on the policy's three-view canvas, with the halved exterior resolution costing 0.14 in hit rate and the multi-view layout 0.12 (Q1; measured on the policy checkpoint's own reasoner, with its base VLM within 0.08 on every view). The named target is far more sharply decodable from its image tokens than from any generator token group (top-cell hit 0.55 against at most 0.39), although the two towers put equal probe mass on the mask, yet the generator's actions follow the object's pixels beyond every matched artifact control and re-target after a simulated displacement (Q2).
- The action head reads the instruction's noun (a swap shifts actions 20.8 cm open-loop; deleting or swapping it costs 39–46 pp of closed-loop success) but does not act on location text: a location sentence moves the actions only 0.5 cm more than a location-free one and toward neither location, and the null holds across guidance scales, under a causal knock-out, on 26 frames with a verified same-class distractor, and in 500 paired closed-loop episodes per condition (Q3).
- A second checkpoint, Cosmos3-Edge-Policy-DROID (a from-scratch dense backbone, not a pretrained VLM; a noun swap shifts its actions 14.5 cm), reproduces the sign and ordering of every pixel contrast and every instruction-versus-location contrast; Nano's 0.5 cm correct-versus-neutral excess does not appear (Appendix K).

Related work is discussed in Appendix A.

2 SETUP

2.1 THE RELEASED POLICY PATHWAY

Cosmos3-Nano-Policy-DROID predicts a 32×8 chunk of absolute joint positions and gripper commands (32 steps at 15 Hz, 2.1 s) from a three-view canvas (wrist view on top at 360×640, two exterior views below at 180×320 each), the proprioceptive state, and a language instruction. It is served through the released framework with 4 UniPC (Zhao et al., 2023) steps and classifier-free guidance 3. Three facts about its inference code frame every experiment. First, the reasoner stream (the "understanding stream" of the Cosmos 3 report) receives only text: the instruction followed by fixed view, duration, and resolution sentences, 91 tokens in total (`_encode_text` embeds text ids only; the visual encoder is reached only through the `generate_reasoner_text` entry point of VLM mode, which the policy server never calls). Second, the canvas is VAE-encoded into a 17×20 clean latent frame in the generator stream (a DiT, which we also call the action head), followed by 8 noisy video frames, a clean state token, and 32 noisy action tokens. Third, the generator attends to the cached text keys and values at every denoising step, never the reverse. This is not an inference-time shortcut: the policy was post-trained with "the official DROID short task instructions as the prompts" and the canvas as the visual observation (NVIDIA, 2026), and the released training pipeline emits only caption text, canvas video, and the action chunk, so the reasoner stream never saw an image during policy training either.

Which reasoner. The checkpoint's reasoner stream and vision encoder can be served on their own in VLM mode (vLLM loads only those tensors), and every headline Q1 and Q2 reasoner measurement (point hits, canvas decomposition, target-named-first probes) uses them; four comparison items were run on Cosmos3-Nano: the question-after-image and swap probe controls (Section 4), the box and quadrant columns of Table 8, and the Cosmos3-Nano probe rows of Table 4. Cosmos3-Nano denotes the released base VLM from which the reasoner stream was initialised, reported as a comparison. Post-training changed the reasoner stream's weights by 2.3% (relative mean absolute difference, 2.2–2.6% in every layer) and the generator stream's by 55%; in the vision encoder only the patch mergers changed by more than 1% (at most 21%; ViT blocks: median 0.006%). Text that a reasoner writes *for* the policy (the 255 nouns below, the swap objects of Section 5, and the closed-loop locations of Section 6) was generated by Cosmos3-Nano.

2.2 EVALUATION DATA

D-DROID. We sample 2,302 episodes from `nvidia/Cosmos3-DROID`, the DROID (Khazatsky et al., 2024) release used for post-training, with at most 5 episodes from each of 520 *scene proxies* (consecutive episodes of one recording session, defined by the release's video-shard triple; the unit of our cluster bootstrap). For each episode we take the first frame after the robot leaves idle and keep the instruction, state, three native 640×360 views, and the ground-truth 32-step chunk. Every frame is thus a training frame and an episode start: the open-loop findings describe the policy in distribution, and the closed-loop simulator (Section 6) is the out-of-distribution test. The target noun is extracted with a regex; the 255 cases where it fails are named by Cosmos3-Nano, and the policy's reasoner names the same noun phrase on 73% of the 168 with ground truth. Hit rates are higher on these frames, as expected when a reasoner chooses the noun (0.78–0.86 against 0.69–0.74 elsewhere for the policy's reasoner, 0.84–0.86 against 0.75–0.77 for Cosmos3-Nano); excluding them changes no conclusion. Ground truth is Grounding DINO (Liu et al., 2024) boxes and SAM (Kirillov et al., 2023) masks per view; 1,001 samples have an exterior-view detection above 0.5. An LLM agent, not a human, checked all 1,001 boxes on contact sheets (protocol in Appendix B): 70.0% correct, 18.6% ambiguous (mostly a part boxed as the whole appliance), 11.4% wrong. Headline numbers use all boxes; on the 701 verified-correct boxes every conclusion is unchanged (Appendix C). Table 3 in Appendix B lists how each analysis subset is derived from the 2,302 frames. As an interface check, the policy's joint MSE to ground truth is 0.048 against 0.118 for holding still, and its endpoint error 11.8 cm against 30.1 cm (all 2,302 frames).

2.3 PAIRED DESIGN AND METRICS

Each condition is run once per sample under that sample's stored noise seed, so a difference between conditions is caused by the condition, not by sampling noise. Our main open-loop metric

is the *endpoint shift*, the Euclidean distance between the end-effector positions (flange plus 10 cm tool offset, via Franka forward kinematics) at the final step of two predicted chunks. Two reference magnitudes calibrate it: changing only the noise seed moves the endpoint by 7.0 cm on average (median 5.8 cm; Appendix D), and changing only the kernel implementation (torch compilation off) by 2.8 cm $[2.4, 3.2]$, a numerical floor below which two runs cannot be told apart. A shift is a magnitude and never shows by itself that actions moved *toward* anything, so we interpret shifts through paired contrasts against a matched control in the same channel and read direction from separate image-plane, ground-truth-endpoint, and simulator tests. All intervals are 95% scene-cluster bootstrap CIs (frames of one scene proxy resampled together; seed 0, 5,000 resamples), regenerated for every table by one script from the released per-sample files; paired tests are Wilcoxon signed-rank (open loop) and exact McNemar (closed loop), and since tiny p-values are routine at $n{=}500$, effect sizes and CIs carry the argument. A second seed reproduces every condition's shift to within 0.3 cm on 100 samples; the correct-vs-neutral contrast, a difference of two such shifts, is 0.7 cm on one seed and 0.2 cm on the other (Appendix D).

3 Q1: The Reasoner Can Find the Object, but Not on the Policy's Canvas

We serve the policy checkpoint's reasoner in VLM mode (vLLM, greedy decoding) and ask for a 2D point for the target on each exterior view, the wrist view, and the full canvas (Table 8 in Appendix F, with Cosmos3-Nano as comparison). A *hit* means the returned point falls inside the target's SAM mask; on the canvas, the point is tested against the mask of whichever view it lands in. An answer without a point counts as a miss: the policy's reasoner answers "There are none." on 2–9% of queries, Cosmos3-Nano never. On single views the reasoner hits 75 and 76% of the time on the exterior views and 71% on the wrist view (78% of the points it returns, on every view; Cosmos3-Nano 76–78%), and lands within 5% of the image diagonal 80–86% of the time. Most misses are ambiguity rather than blindness: the point lands on some same-class detection 83–91% of the time, and on the exterior views hit rate falls from 0.80–0.81 with one detection of the target class to 0.54–0.57 on the 21–22% of frames with several. On the three-view canvas the policy acts on, hit rate drops to 0.46 (Cosmos3-Nano 0.50; paired difference -0.04 $[-0.06, -0.01]$). Decomposing this drop on the 1,001 frames with exterior ground truth (Table 8, bottom) shows two causes of similar size: downscaling the best exterior view to its $180{\times}320$ canvas size lowers the hit rate from 0.76 to 0.61 (-0.14 $[-0.17, -0.11]$); adding the second downscaled view beside it and the wrist view on top gives the policy's canvas, 0.47 on this subset (the 0.46 above is on all 1,169 frames with a detection in any view); three views with native-resolution exteriors score 0.63, so the layout alone costs 0.12 $[0.09, 0.16]$ (Cosmos3-Nano: 0.12 and 0.12). Post-training left this competence nearly intact: per view the two checkpoints' points lie a median 6–8 px apart and agree on hit or miss on 89–92% of frames. The degradation of VLMs on tiled, multi-image inputs (Meng et al., 2024) is here measured on the exact composite the policy consumes.

4 Q2: Who Exposes the Object, and Do the Actions Follow Its Pixels?

Where is the named object linearly decodable? We capture the generator's clean-canvas token states (layers 8, 20, 32) during the released policy call, and the image tokens of the policy checkpoint's reasoner for the same canvas in VLM mode (layers 4–35). For each layer we train a logistic probe from a single token's state to whether that token's $32{\times}32$ cell overlaps the target mask, with frames split 80/20 by scene hash ($n_{\text{test}}{=}176$ frames from held-out scenes; Table 9 in Appendix F). The *top-cell hit* is the fraction of test frames whose highest-scoring cell overlaps the mask. Three controls address the obvious confounds. First, a *position-only* probe that sees only the cell index reaches 0.24 (AUROC 0.69), so most of the distance from chance (0.04) is a spatial prior; appending position features to a token probe changes its hit rate by at most 0.03. Second, the reasoner is causal: with the question after the image ("Where is the {object}?") Cosmos3-Nano's image tokens cannot know the target, and they score 0.21–0.28 across layers 4–35 (Figure 5), below the generator's 0.27–0.35; naming the target *before* the image lifts the policy reasoner's tokens to 0.55 $[0.47, 0.63]$ at layer 20 (0.53 at layer 28; Cosmos3-Nano 0.60 $[0.52, 0.67]$, paired difference -0.05 $[-0.11, 0.02]$), against 0.35 $[0.28, 0.42]$ for the generator at layer 32. Third, an *instruction-swap*

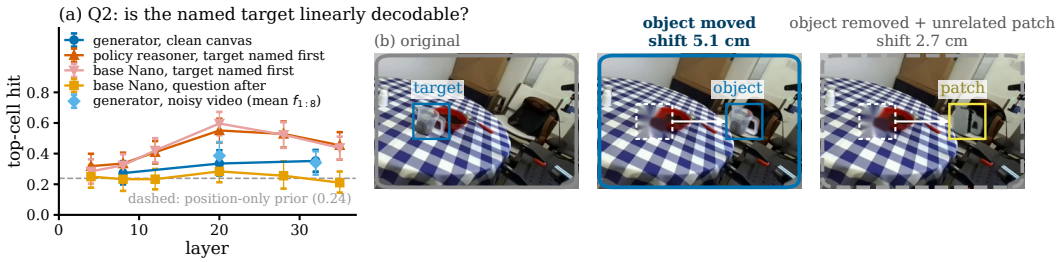


Figure 2: **Q2 in one view. (a)** Top-cell hit of a linear probe for "this canvas cell overlaps the target" by layer, on 176 held-out-scene frames (scene-cluster 95% CIs); per-token noisy-video probes of the generator in Table 11. **(b)** One paired pixel contrast on the median-effect frame ("Pour the contents from the transparent jar into the orange bowl"), cropped to the edit, with this frame's endpoint shifts: the object moved against its artifact-complete control, *patch, object removed*, which inpaints the object and pastes an unrelated patch at the destination. Figures 3 and 4 show the *unrelated patch* control (object left in place) on the same frame; all edits in Figure 4, population statistics in Table 1.

test on Cosmos3-Nano's maps asks whether a readout is about the named target or about objects in general: we recompute the maps with the target noun replaced by another object in the scene (swap prompts of Section 5, n=419). The reasoner's top cell changes on 91% of frames and lands on the swapped object 33% of the time (up from 5%); the generator's map is nearly unchanged (correlation 0.90–0.94) and its top cell lands on the swapped object 7–11% of the time (up from 5–8%). The generator's canvas tokens therefore carry a largely instruction-agnostic object map, the reasoner's a map of the named target (Figure 2a; every layer in Figure 5).

This comparison is between different kinds of tokens (ViT patches in a causal LM against VAE latents in a bidirectional DiT) and establishes linear decodability, not what the generator knows. Further generator probes do not close the gap (Table 11): a 2-layer MLP on the same tokens reaches 0.36 or below, ridge probes from mean-pooled noisy-video, action, and state tokens 0.27–0.30, and per-token probes of the noisy-video maps and of all 33 action-token states at most 0.39 $[0.29, 0.48]$ and 0.28. Top-cell hit rewards a single confident cell, so we also score readouts that do not depend on peakedness, all under one common negative draw (Table 10). Token AUROC is 0.81 for the generator's layer-20 and 32 tokens against 0.69 for the position-only probe and 0.85 for the policy's reasoner; the share of probe mass inside the target mask is 0.10 and 0.09 against 0.05 (mask area 0.04) and 0.11. The target is thus decodable from the generator's tokens, well above the position prior; what separates the towers is mainly where the probe's peak falls. The reasoner's probe puts no more mass on the mask than the generator's at layer 20 (difference 0.01 $[-0.00, 0.02]$) but ranks the mask cells first more often (top-cell hit $+0.19$ $[0.10, 0.28]$; IoU of the k top cells with the k mask cells $+0.10$ $[0.06, 0.15]$). The causal tests that follow, not the probes, arbitrate what the generator uses.

Do actions follow the object's pixels? Probes show that information is present, not that it is used, so we edit 500 frames in the exterior view where the target is largest (Figure 2b; all edits in Figure 4), keeping instruction and seed fixed. *Moved* cuts the object out, inpaints the hole, and pastes it $\pm$160 px (25% of the width) away; *removed* only inpaints; *distractor* pastes a copy and keeps the original. Three controls match the artifacts. *Unrelated patch* pastes a same-sized patch from another scene at the new location and leaves the object in place. *Patch, object removed* inpaints the object *and* pastes the patch, so it differs from the moved edit only in whether the pasted pixels are the object. *Other object moved* applies the full edit to a different object in the same view (the one Cosmos3-Nano named for the swap test of Section 5; n=367).

Table 1 (top) shows that actions respond to the object, not merely to edited pixels. Moving the object shifts the endpoint by 7.7 cm, removing it by 5.7 cm, and adding a copy of it by 5.3 cm, while the unrelated patch shifts it by 3.7 cm, only 0.9 cm above the 2.8 cm numerical floor; the moved edit exceeds the patch by 4.0 cm $[3.4, 4.7]$ (larger on 76% of samples). Moving a *different* object with the identical procedure shifts the endpoint by 4.3 cm, indistinguishable from the unrelated patch on the same 367 frames (4.0 cm; difference 0.4 cm $[-0.1, 0.9]$, p=0.15) and half of moving the target there (8.0 cm; difference 3.7 cm $[2.8, 4.6]$, larger on 75%): the response is to the target, not to an object-shaped edit. The artifact-complete control is more sobering: inpainting the target and pasting a patch

already shifts the endpoint by 6.4 cm, so most of the moved edit's effect is the target vanishing, and its reappearance at the new location adds 1.3 cm $[0.8, 1.8]$ (moved larger on 57%, $p < 0.001$). The effect also has a direction. On the 196 samples with usable data-derived camera models, the projected endpoint moves 6.7 px $[2.8, 11.4]$ along the object's 160 px displacement, against 0.2 px for the patch, with a correlation of 0.23 between the object's move and the endpoint's image-plane move.

Physically moving the object. The single-view paste leaves the other views unchanged and its image-plane metric depends on noisy cameras, so we repeat the test in simulation with all views re-rendered. For the first 200 MolmoSpaces episodes (Section 6) we query the released policy on the first observation with the target at its benchmark pose, displaced by ± 10 cm laterally in the robot base frame, and, as a control, with the nearest other movable object displaced by the same vector, under the same seed (n=169; the other 31 episodes have no movable object 8–60 cm from the target within 15 cm of its height). Moving the target shifts the endpoint by 12.1 cm $[10.9, 13.3]$ ($+10$ cm) and 11.7 cm (-10 cm), of which 6.0 cm $[4.9, 7.0]$ lies along the displacement direction (toward the new position on 79% $[75, 83]$ of trials). Moving the control object also perturbs the chunk (7.3 cm $[6.1, 8.6]$) but with no direction (-0.3 cm $[-1.6, +1.0]$ along the vector, positive on 47%; target vs. control $p < 0.001$). With consistent views the first chunk re-targets by roughly 60% of a 10 cm displacement. Together with the noun swap of Section 5, this says that the generator resolves the named object and acts on it, which is what "knowing where the box is" means for a policy; what it does not do is expose that resolution as a sharply peaked linear map, as the reasoner does.

5 Q3: The Action Head Does Not Act on Location Text

Text interventions. On the same 500 frames we append one sentence to the instruction. *Correct* gives the ground-truth quadrant in the best exterior view ("The towel is in the bottom-left quadrant of the left third-person view"), *wrong* the opposite quadrant, *reasoner* Cosmos3-Nano's quadrant answer, and *neutral* is a length-matched sentence with no location ("The towel is on the table in the left third-person view"). All four shift the endpoint by 4.0–4.5 cm, 1.2–1.7 cm above the numerical floor, and none changes joint MSE to ground truth (paired differences -0.0004 to $+0.0016$, all CIs containing zero; Table 1). The correct sentence's shift exceeds the neutral one's by 0.5 cm $[0.3, 0.8]$, so we test whether it pulls toward the stated location (Table 6, Appendix E). It does not. The correct and wrong chunks differ by 3.8 cm, about as much as either differs from the neutral chunk (3.6 and 3.4 cm), and their shifts point in similar directions (mean cosine 0.49 $[0.45, 0.53]$). In the image plane (n=208) the correct sentence moves the projected endpoint closer to the true quadrant on 45% $[38, 52]$ of frames; camera-free, the endpoint error to the ground-truth chunk is the same under all three sentences (11.3, 11.4, 11.3 cm; correct minus wrong -0.1 cm $[-0.4, 0.2]$). The roughly 4 cm effect is generic sensitivity to the prompt, and not an artifact of the guidance scale: on the same 200 frames at guidance 1, 3, and 5 (Figure 6b) the shift is 2.2–2.4, 3.8–4.4, and 4.8–5.3 cm, the same for all three sentences to within 0.6 cm, and MSE never improves (at guidance 5 the correct sentence makes it worse, $+0.0035$ $[0.0014, 0.0059]$, p=0.025; wrong and neutral $+0.0017$ and $+0.0011$). What remains is a small, content-dependent residual that we cannot explain and that is itself fragile: it is 0.7 cm on the stored seed and 0.2 cm on a second seed (Appendix D), and it does not appear on a second checkpoint (Appendix K). The correct sentence, which differs from the wrong one only in its quadrant words, also shifts the chunk more than the wrong one ($+0.2$ $[0.0, 0.5]$, $+0.3$ $[-0.1, 0.8]$, and $+0.5$ cm $[0.1, 1.0]$ at g=1, 3, and 5; paired contrasts are from unrounded means, Table 5) and has a 0.3 cm $[0.0, 0.5]$ larger component along ground truth than the neutral one (p=0.02), yet it does not lower the error and pulls toward neither location. The excess is larger when the stated quadrant is on the right (0.7 against 0.3 cm; p=0.04), but the wrong sentence, which then says "left", shows no such split, and the excess does not track the released chunk's error (Spearman 0.08; Appendix E): an effect of wording with no location-specific component.

Location text where it is needed: frames and phrasings. The quadrant sentences are redundant on most frames, since the instruction already identifies the target, and they are phrased unlike DROID instructions. We therefore select the frames of the 1,001 on which the chosen exterior view contains a second detection of the target class to one side of the target, with a singular noun that appears verbatim in an instruction without a side phrase (thresholds in Appendix B; n=75, 61 scenes, 42 with a camera model). Here a side phrase is informative. We embed the phrase as a relative

Table 1: **Open-loop interventions** on 500 D-DROID frames, each paired with the released prediction under the same noise seed. Shift: distance between predicted end-effector endpoints (cm, 95% scene-cluster CI); the reference rows give the shift from a numerical change alone and from a seed change alone, on the first 200 and 100 frames. ΔMSE: paired change in joint MSE to the ground-truth chunk (released: 0.045), not applicable to pixel edits, where the ground truth no longer matches the scene. Paired contrast: mean shift difference between the row and the named control, with the fraction of frames on which the row is larger. Italics: rows that serve as a comparison target for another row (references, pixel controls, neutral sentence). Bold: the two headline effects.

Channel	Condition	Shift (cm)	ΔMSE	Paired contrast (cm)
Reference	*kernel implementation changed* (n=200)	2.8 [2.4, 3.2]	–	numerical floor
	noise seed changed (n=100)	7.0 [6.1, 7.8]	–	sampling variation
Pixels	object moved	**7.7** [7.0, 8.5]	–	vs. patch $+4.0$ [3.4, 4.7], 76%
	object removed	5.7 [5.1, 6.3]	–	
	copy of the object added	5.3 [4.7, 5.9]	–	
	unrelated patch	3.7 [3.4, 4.1]	–	
	patch, object removed	6.4 [5.9, 7.1]	–	moved vs. this $+1.3$ [0.8, 1.8], 57%
	other object moved (n=367)	4.3 [3.9, 4.8]	–	vs. patch $+0.4$ [-0.1, 0.9], 54%
Location text	correct location	4.5 [4.1, 4.9]	$+0.001$ [-0.001, 0.004]	vs. neutral $+0.5$ [0.3, 0.8], 55%
	wrong location	4.2 [3.8, 4.6]	$+0.001$ [-0.002, 0.003]	
	reasoner location	4.5 [4.0, 4.9]	$+0.002$ [-0.001, 0.004]	
	neutral sentence	4.0 [3.6, 4.3]	-0.000 [-0.002, 0.001]	
Instruction (positive control)	paraphrase (n=495)	5.6 [5.0, 6.3]	$+0.001$ [-0.001, 0.003]	
	instruction deleted	14.8 [13.5, 16.0]	$+0.016$ [0.011, 0.021]	
	object swapped (n=419)	**20.8** [19.0, 22.5]	$+0.026$ [0.019, 0.033]	
	instruction of another scene	23.6 [21.9, 25.3]	$+0.034$ [0.028, 0.041]	

clause naming either the target's side ("Put the mug *that is on the left* in the sink") or the distractor's side, and also append it as a sentence; as positive controls on the same frames and seeds we delete the instruction and, on the 33 frames that have one, apply the noun swap (every readout in Table 7, Appendix E).

Training distribution. By a rule-based labelling (Appendix B), 24% of the 2,144 distinct D-DROID instructions contain "left" or "right": 10% only as a motion direction ("move the towel to the right") and 11% as an instance selector ("the left drawer"; 9% on the manipulated object, 5% in the postnominal form closest to ours, "the cup on the left"). The 90,123 distinct instructions of the full post-training set (172,917 annotations) give 22% (any), 7% (motion only), 11% (selector), and 6% (postnominal, on the object); our relative clause and appended sentence occur in no D-DROID instruction and in 15 and 19 instructions of the full set.

Result. The two embedded phrasings shift the endpoint by 4.6 and 4.1 cm from the released prediction and by 5.4 cm $[4.4, 6.6]$ from each other, in partly shared rather than opposite directions (cosine 0.27); the controls shift it by 17.5 and 20.7 cm. The phrasings leave joint MSE and the endpoint error unchanged (target minus distractor phrasing: -0.000 $[-0.005, 0.005]$ and -0.0 cm $[-0.9, 0.8]$), while the controls raise the error by 8.2 and 9.3 cm. In the image plane (n=42), naming the distractor instead of the target moves the projected endpoint 1.9 px $[-1.2, 5.1]$ toward it (appended: -1.1 px $[-2.6, 0.2]$) against a 132 px separation. A visual check found a clear second instance of the target class on only 26 of the 75 frames (Appendix B), so the informative test is that subset, with limited power. There the phrasings differ by 6.4 cm $[4.3, 8.9]$ and the endpoint-error contrast is -0.5 cm $[-1.9, 0.7]$ against the $+13.0$ cm the noun swap adds on the same frames. Re-targeting on more than 15% of these frames would therefore have been detected by the swap yardstick, and on more than 24% by the deleted-instruction yardstick. The image-plane contrast (n=15: $+2.4$ px $[-4.1, 8.4]$ against 27.8 px for the swap) excludes only re-targeting on more than 30% of frames (Table 7; the 31 frames with instances on opposite sides of the image center line give the same nulls). Within these limits, a phrase that would let a reader pick the right instance perturbs the chunk like any other sentence and steers it toward neither instance, on frames where deleting or swapping the noun moves it by 17–21 cm.

Positive control: the action head does read the instruction. A null for location text is informative only if the policy uses language at all, so we vary the instruction on the same frames and seeds (Table 1). A paraphrase by another DROID annotator shifts the endpoint by 5.6 cm, only slightly more than the location sentences; deleting the instruction shifts it by 14.8 cm and raises joint MSE by

0.016 $[0.011, 0.021]$; another scene's instruction shifts it by 23.6 cm. Most diagnostic is the *object swap*: Cosmos3-Nano names another movable object in the exterior view, and its name replaces the target noun with every other word unchanged ("Put the *kettle lid* on the kettle" $\rightarrow$ "Put the *apple* on the kettle"; n=419). The swap shifts the endpoint by 20.8 cm, raises MSE by 0.026 $[0.019, 0.033]$, and, on the 175 samples with a camera model, moves the projected endpoint 16.7 px $[10.6, 22.9]$ toward the newly named object, 11% of the mean 147 px between the two. The action head reads the instruction, resolves the named object, and acts on it; its insensitivity to location text is specific to that content.

Where does the action head look, and does it matter? We group the 91 prompt tokens into the *instruction*, the appended *location sentence*, the fixed *meta* sentences, and chat-template tokens; *content text* means the first three. Content text receives 0.32, 0.21, and 0.11 of the action queries' attention mass at layers 0, 4, and 6 but at most 0.012 from layer 8 on (even layers captured), where text mass sits on template tokens acting as an attention sink (n=200; Xiao et al., 2024; Figure 6a). Because attention mass misses indirect routes, we also zero the value vectors of chosen text tokens at chosen layers and measure the endpoint shift from a compile-off reference run (Table 12 in Appendix G); each knock-out has a matched control. Knocking out the location sentence at every layer shifts the endpoint by 4.9 cm $[4.3, 5.7]$, and the neutral sentence, on the same frames, by 4.5 cm $[4.0, 5.1]$ (paired difference 0.4 cm $[-0.1, 0.9]$, p=0.41); both leave MSE unchanged or marginally better (-0.0025 $[-0.0051, -0.0001]$ and -0.0014 $[-0.0036, 0.0007]$). Knocking out the instruction (7.4 cm, MSE worse on 60% of frames, p=0.003) or all content text (16.6 cm, ΔMSE $+0.020$ $[0.012, 0.026]$) degrades the actions. From layer 4 or 8 onward the content knock-out still shifts the endpoint by 12.0 and 11.2 cm with worse MSE, from layer 14 onward only by 3.7 cm with unchanged MSE (p=0.78): text is consumed causally through layer 13 by a route the attention mass does not show, and the location sentence is no more used than a location-free one at any depth.

6 Closed Loop: No Net Effect of Location Text

Open-loop shifts might still matter once errors compound, so we run the released checkpoint in MolmoSpaces (Kim et al., 2026) FrankaPickDroidMiniBench, a MuJoCo benchmark of fixed pick episodes for a DROID-configured Franka in procedurally generated rooms of about 90 posed objects. An episode succeeds if, at its end, the target is lifted and touches only the robot; it ends on success or after 500 control steps (32-step chunks executed as released; 7.6 replanning steps on average). We run the first 500 episode specifications under seven paired prompt conditions (same simulator seed). *Reasoner quadrant* queries Cosmos3-Nano for the target's quadrant on the exterior view at every replanning step and appends the sentence; *wrong quadrant* uses the opposite one; *reasoner point* appends Cosmos3-Nano's normalized point (Appendix B); *mirrored point* reflects it through the image center; two language controls empty or replace the instruction. The point conditions carry the argument: the targets sit near the image center (89% within 0.05 of a quadrant boundary), so a verbal quadrant is right only 53% of the time (Appendix I), whereas the appended point lies within 20 px (3% of the image diagonal) of the nearest projected point of the target on 89% of first replanning steps.

Table 2 shows no net effect of location text. The released pipeline succeeds on 59.2% of episodes; adding the reasoner's point gives 58.8% (paired difference -0.4 pp $[-4.4, +3.4]$, 49 episodes gained and 51 lost, p=0.92), adding its quadrant also 58.8%, and wrong locations do not hurt significantly (-1.0 and -2.8 pp). With 100–112 discordant pairs per contrast, the minimum detectable effect (two-sided McNemar, α=0.05) is roughly 4 pp: the point condition rules out a gain of 3.4 pp or more but not a smaller one. Every location condition flips the outcome of about a fifth of the episodes in both directions, the closed-loop face of the generic 4 cm prompt sensitivity; without a seed-only reference we cannot say how much of this is prompt-specific.

What the point cue can test. The point sentence varies little: the reasoner's first point has x=0.51 and y=0.47 with standard deviation 0.07 each across episodes, although the mirrored point lies a median 65 px (9% of the diagonal) from it and within 20 px of the target (same reference) on only 16% of episodes (the reasoner's point: 89%). Of the 204 baseline failures, all of which run to the time limit, 102 touch the target without holding it and 9 hold and lose it; 46 bring the gripper near it without contact, and only 47 (9% of episodes) never approach it, the class a location cue could

Table 2: **Closed-loop pick success** (500 paired episodes per condition; last two rows: language positive controls). Δ: paired difference from the released pipeline (pp, 95% bootstrap CI); $+/-$: discordant episodes gained or lost; p: exact McNemar.

Prompt condition	Success (%) ↑	Δ (pp)	$+$ / $-$	p
Released pipeline	**59.2** [54.8, 63.4]	–	–	–
+ reasoner point	58.8 [54.4, 63.2]	-0.4 [-4.4, $+3.4$]	49 / 51	0.92
+ mirrored point	56.4 [52.0, 60.8]	-2.8 [-7.0, $+1.2$]	48 / 62	0.21
+ reasoner quadrant	58.8 [54.4, 63.0]	-0.4 [-4.6, $+3.8$]	55 / 57	0.92
+ wrong quadrant	58.2 [53.8, 62.4]	-1.0 [-5.0, $+3.0$]	52 / 57	0.70
instruction deleted	20.0 [16.6, 23.6]	-39.2 [-44.2, -34.2]	24 / 220	< 0.001
instruction of another object	12.8 [9.8, 15.8]	-46.4 [-51.4, -41.4]	19 / 251	< 0.001

repair by redirecting the reach (Appendix I). The closed-loop null is thus a null for a narrow-range cue on a benchmark whose failures are mostly grasping failures.

The benchmark is language-sensitive. The null is not because the benchmark ignores the prompt: emptying the instruction drops success to 20.0% (-39.2 pp [-44.2, -34.2]), and replacing it with another episode's instruction, which names an object that is not the pick target, drops it to 12.8% (-46.4 pp; Table 2, bottom). A prompt that changes the target changes success by 39–46 pp; a prompt that adds the target's location changes it by less than one.

7 Discussion

What the policy is at act time. In the released Cosmos3-Nano-Policy-DROID the inheritance of the VLM's competence that a unified model promises is architectural, not functional: the reasoner stream's share of act-time token passes is 0.4% (Table 3) and its visual encoder is never called. The policy is a video-action DiT conditioned on Qwen3-VL *text* states. Post-training moved the generator stream's weights by 55% but the reasoner stream's by 2.3%, and left the reasoner's grounding within 0.08 of its base VLM's: the policy carries a competent grounder that it does not call. The two towers also fail on different frames (Appendix J). Sharing a backbone does not make the policy use the VLM's grounding; what matters is the channel and whether training required reading it.

A grounding-then-act pipeline on this model needs a channel the action head was trained to read, or training that opens one; neither the prompt (Section 5) nor zero-shot image tokens in the reasoner stream (Appendix H) is such a channel. An adapter should query the reasoner on single native-resolution views (Section 3).

Limitations. We study two checkpoints of one model family (all but the open-loop interventions on the larger only), one embodiment, and one closed-loop benchmark with a single exterior camera, a narrow-range point cue, and mostly grasping failures. All open-loop frames are training frames and episode starts; the noun swap bounds memorisation. The distractor test has 26 verified frames and excludes re-targeting only above about 15%; the correct sentence's 0.5 cm excess is unexplained; closed loop lacks a location-free sentence, a simulator oracle, and a seed-only reference. Probes are linear or shallow; the question-after-image and swap maps, the box and quadrant columns of Table 8, and every reasoner-written prompt come from Cosmos3-Nano. Boxes come from a detector checked by an LLM agent, not a human, and our affine cameras are poor (median inlier fraction 0.5), so every conclusion rests on 3D metrics and same-seed pairing.

8 Conclusion

In Cosmos3-Nano-Policy-DROID, the named object is far more sharply exposed to a linear probe in the policy's own reasoner image tokens than in any generator token group, yet the generator is the tower that resolves the instruction's noun and acts on the object's pixels. At act time the reasoner sees only text, and the released policy does not act on text-borne locations at any depth or guidance scale tested, within the resolution of our designs, while it loses 39–46 pp without the noun. Whether this reflects the channel or captions that rarely select an instance by side (Section 5) is not separated here.

AI USE STATEMENT

We use generative AI to implement code, do results analysis, and polish the paper.

REFERENCES

Guillaume Alain and Yoshua Bengio. Understanding Intermediate Layers Using Linear Classifier Probes, 2017. URL https://arxiv.org/abs/1610.01644. ICLR 2017 Workshop.

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-JEPA 2: Self-Supervised Video Models Enable Understanding, Prediction and Planning, 2025. URL https://arxiv.org/abs/2506.09985.

Pranav Atreya, Karl Pertsch, Tony Lee, Moo Jin Kim, Arhan Jain, Artur Kuramshin, Clemens Eppner, Cyrus Neary, Edward Hu, Fabio Ramos, et al. RoboArena: Distributed Real-World Evaluation of Generalist Robot Policies. In *Conference on Robot Learning (CoRL)*, 2025.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-VL Technical Report, 2025. URL https://arxiv.org/abs/2511.21631.

Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a Vision-Language-Action Model with Open-World Generalization, 2025a. URL https://arxiv.org/abs/2504.16054.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Shi, Laura Smith, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A Vision-Language-Action Flow Model for General Robot Control. In *Robotics: Science and Systems XXI*, RSS2025. Robotics: Science and Systems Foundation, 2025b. doi: 10.15607/rss.2025.xxi.010. URL http://dx.doi.org/10.15607/rss.2025.xxi.010.

Jun Cen, Chaohui Yu, Hangjie Yuan, Yuming Jiang, Siteng Huang, Jiayan Guo, Xin Li, Yibing Song, Hao Luo, Fan Wang, Deli Zhao, and Hao Chen. WorldVLA: Towards Autoregressive Action World Model, 2025. URL https://arxiv.org/abs/2506.21734.

Chi-Lam Cheang, Guangzeng Chen, Ya Jing, Tao Kong, Hang Li, Yifeng Li, Yuxiao Liu, Hongtao Wu, Jiafeng Xu, Yichu Yang, Hanbo Zhang, and Minzhao Zhu. GR-2: A Generative Video-Language-Action Model with Web-Scale Knowledge for Robot Manipulation, 2024. URL https://arxiv.org/abs/2410.06158.

Boyuan Chen, Zhuo Xu, Sean Kirmani, Brian Ichter, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. SpatialVLM: Endowing Vision-Language Models with Spatial Reasoning Capabilities. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14455–14465.

IEEE, 2024. doi: 10.1109/cvpr52733.2024.01370. URL http://dx.doi.org/10.1109/cvpr52733.2024.01370.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, Jiasen Lu, Taira Anderson, Erin Bransom, Kiana Ehsani, Huong Ngo, YenSung Chen, Ajay Patel, Mark Yatskar, Chris Callison-Burch, Andrew Head, Rose Hendrix, Favyen Bastani, Eli VanderBilt, Nathan Lambert, Yvonne Chou, Arnavi Chheda, Jenna Sparks, Sam Skjonsberg, Michael Schmitz, Aaron Sarnat, Byron Bischoff, Pete Walsh, Chris Newell, Piper Wolters, Tanmay Gupta, Kuo-Hao Zeng, Jon Borchardt, Dirk Groeneveld, Crystal Nam, Sophie Lebrecht, Caitlin Wittlif, Carissa Schoenick, Oscar Michel, Ranjay Krishna, Luca Weihs, Noah A. Smith, Hannaneh Hajishirzi, Ross Girshick, Ali Farhadi, and Aniruddha Kembhavi. Molmo and PixMo: Open Weights and Open Data for State-of-the-Art Vision-Language Models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 91–104. IEEE, 2025. doi: 10.1109/cvpr52734.2025.00018. URL http://dx.doi.org/10.1109/cvpr52734.2025.00018.

Danny Driess, Jost Tobias Springenberg, Brian Ichter, Lili Yu, Adrian Li-Bell, Karl Pertsch, Allen Z. Ren, Homer Walke, Quan Vuong, Lucy Xiaoyang Shi, and Sergey Levine. Knowledge Insulating Vision-Language-Action Models: Train Fast, Run Fast, Generalize Better, 2025. URL https://arxiv.org/abs/2505.23705.

Samuel Greydanus, Anurag Koul, Jonathan Dodge, and Alan Fern. Visualizing and Understanding Atari Agents. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1792–1801, 2018. URL https://proceedings.mlr.press/v80/greydanus18a.html.

Bear Häon, Kaylene Stocking, Ian Chuang, and Claire Tomlin. Mechanistic Interpretability for Steering Vision-Language-Action Models, 2025. URL https://arxiv.org/abs/2509.00328.

John Hewitt and Percy Liang. Designing and Interpreting Probes with Control Tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 2733–2743. Association for Computational Linguistics, 2019. doi: 10.18653/v1/d19-1275. URL http://dx.doi.org/10.18653/v1/d19-1275.

Jonathan Ho and Tim Salimans. Classifier-Free Diffusion Guidance, 2021. URL https://arxiv.org/abs/2207.12598. NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications.

Yucheng Hu, Yanjiang Guo, Pengchao Wang, Xiaoyu Chen, Yen-Jen Wang, Jianke Zhang, Koushil Sreenath, Chaochao Lu, and Jianyu Chen. Video Prediction Policy: A Generalist Robot Policy with Predictive Visual Representations, 2025. URL https://arxiv.org/abs/2412.14803.

Joel Jang, Seonghyeon Ye, Zongyu Lin, Jiannan Xiang, Johan Bjorck, Yu Fang, Fengyuan Hu, Spencer Huang, Kaushil Kundalia, Yen-Chen Lin, Loic Magne, Ajay Mandlekar, Avnish Narayan, You Liang Tan, Guanzhi Wang, Jing Wang, Qi Wang, Yinzhen Xu, Xiaohui Zeng, Kaiyuan Zheng, Ruijie Zheng, Ming-Yu Liu, Luke Zettlemoyer, Dieter Fox, Jan Kautz, Scott Reed, Yuke Zhu, and Linxi Fan. DreamGen: Unlocking Generalization in Robot Learning through Video World Models, 2025. URL https://arxiv.org/abs/2505.12705.

Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Srirama, Lawrence Chen, Kirsty Ellis, Peter Fagan, Joey Hejna, Masha Itkina, Marion Lepert, Yecheng Ma, Patrick Miller, Jimmy Wu, Suneel Belkhale, Shivin Dass, Huy Ha, Arhan Jain, Abraham Lee, Youngwoon Lee, Marius Memmel, Sungjae Park, Ilija Radosavovic, Kaiyuan Wang, Albert Zhan, Kevin Black, Cheng Chi, Kyle Hatch, Shan Lin, Jingpei Lu, Jean Mercat, Abdul Rehman, Pannag Sanketi, Archit Sharma, Cody Simpson, Quan Vuong, Homer Walke, Blake Wulfe, Ted Xiao, Jonathan Yang, Arefeh Yavary, Tony Zhao, Christopher Agia, Rohan Baijal, Mateo Castro, Daphne Chen, Qiuyu Chen, Trinity Chung, Jaimyn Drake, Ethan Foster, Jensen Gao, David Herrera, Minho Heo, Kyle

Hsu, Jiaheng Hu, Donovon Jackson, Charlotte Le, Yunshuang Li, Roy Lin, Zehan Ma, Abhiram Maddukuri, Suvir Mirchandani, Daniel Morton, Tony Nguyen, Abigail O'Neill, Rosario Scalise, Derick Seale, Victor Son, Stephen Tian, Emi Tran, Andrew Wang, Yilin Wu, Annie Xie, Jingyun Yang, Patrick Yin, Yunchu Zhang, Osbert Bastani, Glen Berseth, Jeannette Bohg, Ken Goldberg, Abhinav Gupta, Abhishek Gupta, Dinesh Jayaraman, Joseph Lim, Jitendra Malik, Roberto Martín-Martín, Subramanian Ramamoorthy, Dorsa Sadigh, Shuran Song, Jiajun Wu, Michael Yip, Yuke Zhu, Thomas Kollar, Sergey Levine, and Chelsea Finn. DROID: A Large-Scale In-The-Wild Robot Manipulation Dataset. In *Robotics: Science and Systems XX*, RSS2024. Robotics: Science and Systems Foundation, 2024. doi: 10.15607/rss.2024.xx.120. URL http://dx.doi.org/10.15607/rss.2024.xx.120.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-Tuning Vision-Language-Action Models: Optimizing Speed and Success, 2025. URL https://arxiv.org/abs/2502.19645.

Yejin Kim, Wilbert Pumacay, Omar Rayyan, Max Argus, Winson Han, Eli Vanderbilt, Jordi Salvador, Abhay Deshpande, Rose Hendrix, Snehal Jauhri, Shuo Liu, Nur Shafiullah, Maya Guru, Ainaz Eftekhar, Karen Farley, Donovan Clay, Jiafei Duan, Arjun Guru, Piper Wolters, Alvaro Herrasti, Ying-Chun Lee, Georgia Chalvatzaki, Yuchen Cui, Ali Farhadi, Dieter Fox, and Ranjay Krishna. MolmoSpaces: Large-Scale Open Ecosystem for Robot Manipulation and Navigation. In *Robotics: Science and Systems XXII*, RSS2026. Robotics: Science and Systems Foundation, 2026. doi: 10.15607/rss.2026.xxii.091. URL http://dx.doi.org/10.15607/rss.2026.xxii.091.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment Anything. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3992–4003. IEEE, 2023. doi: 10.1109/iccv51070.2023.00371. URL http://dx.doi.org/10.1109/iccv51070.2023.00371.

Jason Lee, Jiafei Duan, Haoquan Fang, Yuquan Deng, Shuo Liu, Boyang Li, Bohan Fang, Jieyu Zhang, Yi Ru Wang, Sangho Lee, Winson Han, Wilbert Pumacay, Angelica Wu, Rose Hendrix, Karen Farley, Eli VanderBilt, Ali Farhadi, Dieter Fox, and Ranjay Krishna. MolmoAct: Action Reasoning Models that can Reason in Space, 2025. URL https://arxiv.org/abs/2508.07917.

Shuang Li, Yihuai Gao, Dorsa Sadigh, and Shuran Song. Unified Video Action Model. In *Robotics: Science and Systems XXI*, RSS2025. Robotics: Science and Systems Foundation, 2025. doi: 10.15607/rss.2025.xxi.074. URL http://dx.doi.org/10.15607/rss.2025.xxi.074.

Weixin Liang, Lili Yu, Liang Luo, Srinivasan Iyer, Ning Dong, Chunting Zhou, Gargi Ghosh, Mike Lewis, Wen-tau Yih, Luke Zettlemoyer, and Xi Victoria Lin. Mixture-of-Transformers: A Sparse and Scalable Architecture for Multi-Modal Foundation Models, 2025. URL https://arxiv.org/abs/2411.04996.

Yue Liao, Pengfei Zhou, Siyuan Huang, Donglin Yang, Shengcong Chen, Yuxin Jiang, Yue Hu, Jingbin Cai, Si Liu, Jianlan Luo, Liliang Chen, Shuicheng Yan, Maoqing Yao, and Guanghui Ren. Genie Envisioner: A Unified World Foundation Platform for Robotic Manipulation, 2025. URL https://arxiv.org/abs/2508.05635.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. Grounding DINO: Marrying DINO with Grounded Pre-training for Open-Set Object Detection. In *Computer Vision – ECCV 2024*, pp. 38–55. Springer Nature Switzerland, 2024. ISBN 9783031729706. doi: 10.1007/978-3-031-72970-6_3. URL http://dx.doi.org/10.1007/978-3-031-72970-6_3.

Hong Lu, Hengxu Li, Prithviraj Singh Shahani, Stephanie Herbers, and Matthias Scheutz. Probing a Vision-Language-Action Model for Symbolic States and Integration into a Cognitive Architecture. In *2025 IEEE International Conference on AI and Data Analytics (ICAD)*, pp. 1–8. IEEE, 2025. doi: 10.1109/icad65464.2025.11114062. URL http://dx.doi.org/10.1109/icad65464.2025.11114062.

Corey Lynch and Pierre Sermanet. Language Conditioned Imitation Learning over Unstructured Data. In *Robotics: Science and Systems XVII*, 2021. doi: 10.15607/rss.2021.xvii.047. URL http://dx.doi.org/10.15607/rss.2021.xvii.047.

Oier Mees, Lukas Hermann, and Wolfram Burgard. What Matters in Language Conditioned Robotic Imitation Learning over Unstructured Data. *IEEE Robotics and Automation Letters*, 7(4):11205–11212, 2022a. URL https://arxiv.org/abs/2204.06252.

Oier Mees, Lukas Hermann, Erick Rosete-Beas, and Wolfram Burgard. CALVIN: A Benchmark for Language-Conditioned Policy Learning for Long-Horizon Robot Manipulation Tasks. *IEEE Robotics and Automation Letters*, 7(3):7327–7334, 2022b. URL https://arxiv.org/abs/2112.03227.

Fanqing Meng, Jin Wang, Chuanhao Li, Quanfeng Lu, Hao Tian, Jiaqi Liao, Xizhou Zhu, Jifeng Dai, Yu Qiao, Ping Luo, Kaipeng Zhang, and Wenqi Shao. MMIU: Multimodal Multi-image Understanding for Evaluating Large Vision-Language Models, 2024. URL https://arxiv.org/abs/2408.02718.

Marco Molinari, Leonardo Nevali, Saharsha Navani, and Omar G. Younis. Emergent World Representations in OpenVLA, 2025. URL https://arxiv.org/abs/2509.24559.

NVIDIA. Cosmos 3: Omnimodal World Models for Physical AI, 2026. URL https://arxiv.org/abs/2606.02800.

NVIDIA, Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi "Jim" Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, Joel Jang, Zhenyu Jiang, Jan Kautz, Kaushil Kundalia, Lawrence Lao, Zhiqi Li, Zongyu Lin, Kevin Lin, Guilin Liu, Edith Llontop, Loic Magne, Ajay Mandlekar, Avnish Narayan, Soroush Nasiriany, Scott Reed, You Liang Tan, Guanzhi Wang, Zu Wang, Jing Wang, Qi Wang, Jiannan Xiang, Yuqi Xie, Yinzhen Xu, Zhenjia 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, 2025. URL https://arxiv.org/abs/2503.14734.

Nikaash Puri, Sukriti Verma, Piyush Gupta, Dhruv Kayastha, Shripad Deshmukh, Balaji Krishnamurthy, and Sameer Singh. Explain Your Move: Understanding Agent Actions Using Specific and Relevant Feature Attribution. In *International Conference on Learning Representations (ICLR)*, 2020.

Yuqi Wang, Xinghang Li, Wenxuan Wang, Junbo Zhang, Yingyan Li, Yuntao Chen, Xinlong Wang, and Zhaoxiang Zhang. Unified Vision-Language-Action Model, 2025. URL https://arxiv.org/abs/2506.19850.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient Streaming Language Models with Attention Sinks, 2024. URL https://arxiv.org/abs/2309.17453.

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 for Robotics, 2024. URL https://arxiv.org/abs/2406.10721.

Michał Zawalski, William Chen, Karl Pertsch, Oier Mees, Chelsea Finn, and Sergey Levine. Robotic Control via Embodied Chain-of-Thought Reasoning, 2025. URL https://arxiv.org/abs/2407.08693.

Matthew D. Zeiler and Rob Fergus. Visualizing and Understanding Convolutional Networks. In *Computer Vision – ECCV 2014*, pp. 818–833. Springer International Publishing, 2014. ISBN 9783319105901. doi: 10.1007/978-3-319-10590-1_53. URL http://dx.doi.org/10.1007/978-3-319-10590-1_53.

Wenliang Zhao, Lujia Bai, Yongming Rao, Jie Zhou, and Jiwen Lu. UniPC: A Unified Predictor-Corrector Framework for Fast Sampling of Diffusion Models. In *Advances in Neural Information Processing Systems 36*, NeurIPS 2023, pp. 49842–49869. Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2023. doi: 10.52202/075280-2170. URL http://dx.doi.org/10.52202/075280-2170.

Ruijie Zheng, Yongyuan Liang, Shuaiyi Huang, Jianfeng Gao, Hal Daumé III, Andrey Kolobov, Furong Huang, and Jianwei Yang. TraceVLA: Visual Trace Prompting Enhances Spatial-Temporal Awareness for Generalist Robotic Policies, 2025. URL https://arxiv.org/abs/2412.10345.

Chunting Zhou, Lili Yu, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the Next Token and Diffuse Images with One Multi-Modal Model, 2024. URL https://arxiv.org/abs/2408.11039.

Xueyang Zhou, Yangming Xu, Guiyao Tie, Yongchao Chen, Guowen Zhang, Duanfeng Chu, Pan Zhou, and Lichao Sun. LIBERO-PRO: Towards Robust and Fair Evaluation of Vision-Language-Action Models Beyond Memorization, 2025. URL https://arxiv.org/abs/2510.03827.

A Related Work

Unified VLM-plus-generator policies. One line of work attaches a flow-matching or diffusion action expert to a pretrained VLM (Black et al., 2025a;b; NVIDIA et al., 2025); another predicts future video jointly with actions, from GR-2 (Cheang et al., 2024) and Video Prediction Policy (Hu et al., 2025) to unified video-action models (Li et al., 2025; Wang et al., 2025; Cen et al., 2025) and video world models used as data engines or planners (Jang et al., 2025; Liao et al., 2025; Assran et al., 2025). Cosmos 3 (NVIDIA, 2026) merges both lines in a Mixture-of-Transformers (Liang et al., 2025) with Transfusion-style (Zhou et al., 2024) autoregressive and diffusion streams, where the diffusion stream reads the autoregressive stream but not the reverse. These papers report task success; we ask which tower carries a specific capability. Closest to us, Driess et al. (2025) show that action-expert gradients can erase backbone knowledge; they ask how to train the interface, we ask what crosses it at inference time.

Spatial cues and language use in robot policies. Several methods train the policy to consume explicit spatial intermediates: points (Yuan et al., 2024), visual traces (Zheng et al., 2025), chains of thought with boxes (Zawalski et al., 2025), depth-aware action reasoning (Lee et al., 2025), or spatially trained VLMs (Chen et al., 2024; Deitke et al., 2025); we ask whether a policy not trained with such cues uses a correct cue in the prompt. That a language-conditioned policy can ignore much of its instruction was measured before VLAs (Lynch & Sermanet, 2021), and Mees et al. (2022a) quantify on CALVIN (Mees et al., 2022b) which design choices make the policy read it at all. Since VLAs likewise often succeed with a corrupted instruction because vision alone predicts the task (Zhou et al., 2025), and FiLM-style injection was needed to make an OpenVLA-based policy follow language at all (Kim et al., 2025), every null here is paired with a positive control that varies the instruction's noun on the same frames.

Probing and intervening on learned policies. Linear probes (Alain & Bengio, 2017) with control tasks (Hewitt & Liang, 2019) ask what a representation encodes; occlusion-style interventions (Zeiler & Fergus, 2014) and perturbation saliency for control policies (Greydanus et al., 2018; Puri et al., 2020) ask what a prediction depends on. Probes have been applied to VLAs (Lu et al., 2025; Molinari et al., 2025), and Häon et al. (2025) steer a VLA by editing activations; our knockout (Section 5) uses the same access in the other direction, removing a pathway to measure what the actions lose. We combine the two families and guard, with a location-free sentence and a guidance sweep, against the confound that classifier-free guidance (Ho & Salimans, 2021) amplifies any prompt difference.

B Implementation Details

Canvas and policy call. The canvas follows the released server layout: wrist view at 360×640 on top and bilinearly resized exterior views (180×320 each) below, padded to 544×640 and encoded to a 17×20 latent grid with 32×32 px cells. Inference uses 4 UniPC steps, guidance 3, and the second (unconditional) pass uses a 10-token empty prompt. The text key-value cache is computed once per call.

Reasoner prompts (Q1, closed loop). The policy checkpoint's reasoner answered the Q1 point prompt (Cosmos3-Nano too, for comparison); Cosmos3-Nano answered the quadrant prompt of Table 8, both closed-loop prompts, and the noun-swap and regex-failure noun queries. Point: "Point to the {object}. Return a json in the format [{"point_2d": [x, y], "label": "{object}"}] with coordinates normalized to 0-1000." Quadrant: "Which quadrant of the image contains the {object}? Answer with exactly one of: top-left, top-right, bottom-left, bottom-right." Decoding is greedy.

Pixel edits. Edits are applied to the exterior view where the target mask is largest, restricted to masks covering 0.2–12% of the image. Holes are filled with Telea inpainting (radius 5). The shift is ± 160 px, with its sign chosen to stay in frame. Only one view is edited per sample.

Probes. Token probes use logistic regression ($C{=}0.05$, balanced class weights) on individual token states, trained on all positive tokens and 5 random negatives per positive; a token is positive if the target mask covers more than 5% of its $32{\times}32$ cell (16.4 positive tokens per frame on average, median 7, of 340; 16,406 positives in total), and the canvas mask is the union of the per-view masks placed at canvas scale. Splits are by scene hash (825 train / 176 test frames from disjoint scenes). The MLP probe has one hidden layer of 256 GELU units with dropout 0.2, trained with AdamW for up to 30 epochs and selected by AUROC on a held-out tenth of the training tokens. Pooled probes use ridge regression with α chosen from $\{10^2, \ldots, 10^5\}$ by top-cell hit on a held-out fifth of the training scenes.

Distractor frames (Section 5). From the 1,001 frames with exterior ground truth we keep those whose chosen exterior view (the one with the higher detector score) contains at least one further detection of the target class with score above 0.3, box IoU below 0.3 with the target's box, and area between 0.15 and 6 times the target's, such that every such detection lies at least 32 px to the same side of the target's box center; the target noun must be singular and appear verbatim in the instruction. This gives 85 frames; we drop the 10 whose instruction already contains "on the left" or "on the right", where an embedded side phrase would double or contradict the existing one (a first version of this experiment included them and produced prompts such as "the orange ring on the left on the left"), leaving 75. The embedded phrase replaces the first occurrence of the noun phrase by "{noun} that is on the {left|right}"; the appended sentence is "The {noun} is the one on the {side} in the {left|right} third-person view." Sides are relative to the nearest distractor's box center; on 31 frames the target and that distractor also lie on opposite sides of the image's vertical center line. Positive controls reuse the deleted-instruction prompt on all 75 frames and Cosmos3-Nano's noun-swap prompt on the 33 frames that have one, run under the same seeds. The image-plane test projects each chunk's endpoint with the frame's affine camera and measures its displacement along the unit vector from the target's box center to the nearest distractor's; for the swap control, along the vector to the swapped object's point. We inspected the 75 target and distractor boxes on contact sheets (the same LLM-agent check as for the ground truth) and record, per frame, whether the distractor is a clear second instance of the target class (26), unclear (11), or not one (38: 7 parts of the target such as a microwave door or a faucet handle, 7 robot clamps, grippers, or camera mounts, and 24 objects of another class or colour); Section 5 reports the verified subset separately.

Side-word labelling (Section 5). Each occurrence of "left" or "right" in an instruction is labelled by ordered regular expressions on its surface form (`src/loop3_droid_text_stats.py`; per-instruction labels released): *motion* when it names a direction of motion ("to the left", "from left to right"), *region* when it names a part of a surface ("on the left side of the counter"), and *selector* when it modifies a noun prenominally ("the left cup", "top right cabinet door", "second drawer from the right") or postnominally ("the cup on the left", "the bin to the right"); a selector is attributed to the manipulated object when it precedes the instruction's first destination cue ("in the", "on top of the"). The rules are approximate (44 of the 617 side words in D-DROID, 7%, match no rule, and a few destinations are labelled as selectors), so the proportions in Section 5 are rounded. Instructions are lowercased and deduplicated (2,144 of the 2,302 frames' instructions are distinct); the full-set proportions are over the 90,123 distinct lowercased strings among the 172,917 annotations of the 57,639 success episodes of `nvidia/Cosmos3-DROID`.

Attention mass. Attention from action queries is recomputed from each decoder layer's input using that layer's own projections, normalizations, and rotary embeddings, averaged over heads,

Table 3: **Sample inclusion.** How each analysis subset derives from the 2,302 D-DROID frames (520 scene proxies).

Subset	n	Criterion
D-DROID	2,302	first moving frame of a Cosmos3-DROID success episode, ≤ 5 per scene proxy
Q1 per view (ext. 1 / ext. 2 / wrist)	735 / 719 / 518	Grounding-DINO detection of the target noun in that view with score >0.5
Q1 canvas	1,169	detection in *any* of the three views (the canvas contains all three)
Ground truth for probes, failure predictor (Appendix J)	1,001	detection in at least one exterior view (825 train / 176 test samples, split by scene hash)
Verified-correct subset	701	box judged correct in the check of the 1,001 exterior boxes (579 train / 122 test)
Q1 canvas decomposition	1,001	the frames with exterior ground truth, queried on the native best exterior view and four re-composed inputs
Intervention frames (pixel, text, language)	500	first 500 of the 1,001 by episode id whose exterior mask covers 0.2–12% of the view (306 scenes)
with camera model	196 / 208 / 175	affine camera with ≥ 12 inliers and inlier fraction ≥ 0.5 in the edited view (pixel edits) / in the best exterior view (text) / among the swap frames
object swap	419	Cosmos3-Nano named a second object not mentioned in the instruction
Attention, guidance sweep, knock-outs	200	first 200 of the 500 intervention frames
Same-class distractor frames	75	of the 1,001: a second detection of the target class ≥ 32 px to one side of the target in the chosen exterior view, singular noun present verbatim in an instruction without a side phrase (61 scenes; 42 with camera model; 26 with a visually verified second instance; 33 with a swap prompt)
Act-time token passes per call	91 / 24,744	reasoner stream: 91 text tokens encoded once; generator stream: $9\times17\times20$ video and 33 action tokens through 4 steps $\times$ 2 passes (reasoner share 0.4%)

and grouped by key type. The reported profile uses the conditional pass at the last denoising step; the other three steps give the same profile (Section 5).

Closed loop. The point sentence reads "The {object} is at position (x, y) of the left third-person view, where $(0, 0)$ is the top-left corner and $(1, 1)$ is the bottom-right corner." MolmoSpaces FrankaPickDroidMiniBench (`json_benchmark_20251231`), first 500 episode specifications, 4 simulation workers, and a custom policy client that sends the native-resolution wrist view and the benchmark's single exterior view, which the released server places in both exterior canvas slots. Success is read at episode end. Paired CIs resample episodes.

Box verification (Section 2). Each of the 1,001 exterior-view detections was drawn on a contact-sheet tile showing the view the detection comes from, the detector's box, and the target noun; an LLM agent inspected every tile and labelled the box *correct*, *ambiguous* (mostly a part boxed as the whole appliance), or *wrong*. The same pass judged 8 episodes outside the 1,001-frame index (1,009 judged), which no analysis uses. The distractor boxes of Section 5 were checked the same way.

C Verified-Subset Analyses

Table 4 repeats the headline grounding numbers on the 701 samples whose exterior-view box was judged correct (Section 2). Every conclusion is unchanged. The policy reasoner's point hit rates rise by 0.02–0.07 (Cosmos3-Nano's by 0.03–0.04) and the probes by more: the generator's layer-20 top-cell hit rises from 0.34 to 0.42 and Cosmos3-Nano's reasoner from 0.60 to 0.64, so the reasoner-generator gap narrows from 0.26 to 0.22 and the two CIs still do not overlap. For the headline pair we refit the policy reasoner's probe on the subset as well, with one common negative draw for it and the generator as in Table 10 (`src/loop5_final.py`): the policy reasoner reaches 0.63 $[0.54, 0.72]$ against 0.42 for the generator, a gap of 0.21 $[0.11, 0.33]$ against 0.19 $[0.10, 0.28]$ on all

ground truth. The verifier was an LLM agent, and its own error rate was not measured against a human; a human check of a random subset is the natural next step and was not done here.

Table 4: Headline grounding numbers on all ground truth vs. the samples whose box was verified correct (701 frames, 579 train / 122 test; point-hit rows on the 525, 527, 265, and 701 of them with ground truth in exterior 1, exterior 2, the wrist view, and any view). Probe rows use the target-named-first reasoner prompt of Table 9, refit on the subset; the Cosmos3-Nano and generator rows come from the base-checkpoint probe run, the last two rows from a common-negative-draw refit (Table 10) that adds the policy reasoner.

Metric	All GT	Verified subset
Policy reasoner point hit, exterior 1 / exterior 2	0.75 / 0.76	0.79 / 0.78
Policy reasoner point hit, wrist / canvas	0.71 / 0.46	0.75 / 0.53
Cosmos3-Nano point hit, exterior 1 / exterior 2	0.78 / 0.76	0.81 / 0.79
Cosmos3-Nano point hit, wrist / canvas	0.78 / 0.50	0.82 / 0.54
Cosmos3-Nano box IoU>0.5, exterior 1 / exterior 2	0.75 / 0.75	0.80 / 0.82
Token probe top-cell hit, position-only prior	0.24 [0.17, 0.31]	0.23 [0.15, 0.31]
Token probe top-cell hit, generator L20 / L32	0.34 / 0.35 [0.28, 0.42]	0.42 / 0.40 [0.31, 0.50]
Token probe top-cell hit, Cosmos3-Nano reasoner L20 / L28 (target named first)	0.60 [0.52, 0.67] / 0.52	0.64 [0.56, 0.72] / 0.53
Token AUROC, generator L20 / Cosmos3-Nano reasoner L20	0.81 / 0.86	0.86 / 0.88
Policy reasoner L20 top-cell hit (common-draw refit)	0.55 [0.47, 0.63]	0.63 [0.54, 0.72]
Policy reasoner minus generator L20 (common-draw refit)	$+0.19$ [0.10, 0.28]	$+0.21$ [0.11, 0.33]

D Seed Replication and Guidance Sweep

Table 5 reports the second-seed replication (first 100 intervention samples, seed$+1$) and the guidance sweep referenced in Sections 2 and 5. Changing only the seed, with frame and prompt fixed, moves the endpoint by 7.0 cm on average (median 5.8 cm). Pixel effects reproduce to within 0.1 cm and text effects to within 0.3 cm across seeds; the correct-vs-neutral gap is 0.7 cm on seed 1 and 0.2 cm on seed 2. Correct $\geq$ wrong $\approx$ neutral holds at every guidance scale (correct minus neutral 0.2 $[-0.0, 0.4]$, 0.6 $[0.2, 1.1]$, and 0.3 $[-0.1, 0.7]$ cm at g=1, 3, 5); the excess is not largest at g=5, but only there does the correct sentence worsen MSE ($+0.0035$ $[0.0014, 0.0059]$ against $+0.0017$ and $+0.0011$ for wrong and neutral text).

Table 5: **Seed check and guidance sweep.** Left: endpoint shift (cm) under the stored seed and a second seed on the first 100 intervention frames. Right: endpoint shift of the three text conditions at guidance 1, 3, and 5 on the same first 200 intervention frames (scene-cluster 95% CIs); these are the bars of Figure 6b. Paired contrasts quoted in the text are computed from unrounded means, so they can differ by 0.1 from the difference of the printed means.

	Seed check (n=100)		Guidance (n=200)		
Condition	seed 1	seed 2	g=1	g=3 (released)	g=5
Object moved	7.1	7.1	–	–	–
Unrelated patch	3.1	3.1	–	–	–
Correct location text	4.6	4.3	2.4 [2.1, 2.9]	4.4 [3.7, 5.1]	5.3 [4.6, 6.0]
Wrong location text	4.1	4.1	2.2 [1.8, 2.7]	4.0 [3.5, 4.6]	4.8 [4.2, 5.4]
Neutral text	3.9	4.1	2.2 [1.8, 2.7]	3.8 [3.2, 4.4]	4.9 [4.2, 5.7]

E Location-Text Direction and Distractor Tests

Table 6 gives the directional readouts of the location sentences on the 500 intervention frames (Section 5), computed from the stored per-sample predictions: the direct contrast between the correct and wrong chunks, the image-plane pull toward the stated quadrant, and camera-free endpoint errors to the ground-truth chunk endpoint. Direct contrasts between the chunks: correct and wrong are 3.8 cm $[3.5, 4.1]$ apart, correct and neutral 3.6 $[3.3, 3.9]$, wrong and neutral 3.4 $[3.1, 3.7]$; the cosine between the correct and wrong shifts is 0.49 $[0.45, 0.53]$; the paired endpoint error, correct minus wrong, is -0.1 cm $[-0.4, 0.2]$ and correct minus neutral -0.0 $[-0.2, 0.2]$; along ground truth, correct minus neutral is $+0.3$ $[0.0, 0.5]$ (p=0.02) and correct minus wrong $+0.2$ $[-0.1, 0.5]$. The correct sentence's residual excess (Section 5; `src/loop4_w4_text_residual.py`) is

not carried by a view word (left vs. right third-person view: 0.6 and 0.5 cm, p=0.97) or a vertical word (top 0.5, bottom 0.6 cm), is larger for a stated right-hand quadrant (0.7 $[0.4, 1.1]$ against 0.3 $[-0.1, 0.6]$ cm, p=0.04; at g=1 0.5 against -0.2 cm, p=0.02) without a matching split for the wrong sentence, and correlates only weakly with the released chunk's endpoint error (Spearman 0.08, p=0.07). Table 7 gives every readout of the same-class-distractor test. On the 26 verified frames the distractor-side phrasing raises the endpoint error by 1.3 cm $[0.2, 2.6]$ and the target-side phrasing by 0.8 cm $[-0.2, 1.6]$ (paired difference -0.5 cm $[-1.9, 0.7]$, p=0.82), both an order of magnitude below the swap control on the same frames ($+13.0$ cm); with 26 frames we cannot tell a 0.5 cm re-targeting component from noise, which is the power limit stated in Section 5.

Table 6: **Direction of the location-sentence shifts** (n=500 frames; image-plane columns on the 208 with a camera model; scene-cluster 95% CIs). Error: end-effector endpoint error to the ground-truth chunk endpoint; Δ: paired change from the released prediction (11.1 cm). Closer: fraction of frames on which the sentence moves the projected endpoint closer to the stated quadrant's center (for the neutral sentence, the true quadrant); cosine: between the image-plane shift and the wrong-to-true direction (sign flipped for the wrong sentence so that positive means toward its stated quadrant). Along GT: component of the 3D shift along the direction from the released endpoint to the ground-truth endpoint. Direct contrasts in the text of Appendix E.

Sentence	Shift (cm)	Error (cm)	Δ error (cm)	Closer to stated (%)	Cosine toward stated	Along GT (cm)
correct location	4.5 [4.1, 4.9]	11.3 [10.6, 12.0]	$+0.2$ [-0.1, 0.6]	45 [38, 52]	-0.04 [-0.14, 0.07]	0.9 [0.6, 1.3]
wrong location	4.2 [3.8, 4.6]	11.4 [10.6, 12.2]	$+0.3$ [0.0, 0.6]	53 [47, 60]	$+0.09$ [-0.01, 0.18]	0.7 [0.4, 1.1]
neutral sentence	4.0 [3.6, 4.3]	11.3 [10.6, 12.1]	$+0.3$ [-0.0, 0.6]	50 [44, 57]	-0.00 [-0.10, 0.10]	0.7 [0.4, 1.0]

Table 7: **Same-class-distractor test** (75 frames, 61 scenes, same seeds as the released prediction; scene-cluster 95% CIs). Shift: endpoint distance from the released prediction; ΔMSE and Δ error: paired changes in joint MSE and in endpoint error to the ground-truth chunk endpoint (released: 0.045 and 10.8 cm); px toward distractor: signed projected-endpoint displacement along the target-to-distractor direction on the 42 frames with a camera model (positive = toward the distractor); px, control axis: for the deleted instruction the unsigned image-plane shift, for the swap the signed displacement toward the swapped object (n=16). Contrast rows are paired differences between the target and distractor phrasings. Subset rows repeat both contrasts on the 26 frames whose distractor is a visually verified second instance (15 with a camera model) and on the 31 whose target and distractor lie on opposite sides of the image center line (18 with a camera model); the verified subset's swap control adds $+13.0$ cm $[3.6, 22.5]$ of error (n=12) and 27.8 px toward the swapped object (n=5).

	Condition	Shift (cm)	ΔMSE	Δ error (cm)	px toward distractor	px, control axis
Side phrase	embedded, target's side	4.6 [3.7, 5.8]	-0.000 [-0.005, 0.005]	$+0.8$ [-0.1, 1.6]	-0.1 [-2.8, 2.6]	–
	embedded, distractor's side	4.1 [3.3, 5.0]	-0.000 [-0.006, 0.005]	$+0.8$ [0.1, 1.5]	$+1.8$ [-0.7, 4.7]	–
	appended, target's side	3.7 [3.0, 4.7]	$+0.000$ [-0.003, 0.004]	$+0.4$ [-0.2, 1.0]	$+1.2$ [-0.5, 3.1]	–
	appended, distractor's side	3.7 [3.1, 4.4]	$+0.002$ [-0.002, 0.006]	$+0.4$ [-0.2, 1.1]	$+0.1$ [-1.5, 1.8]	–
Positive control	instruction deleted	**17.5** [15.0, 20.2]	$+0.024$ [0.013, 0.035]	$+8.2$ [5.7, 10.8]	–	30.2 [22.1, 39.6]
	noun swapped (n=33)	**20.7** [15.3, 26.1]	$+0.047$ [0.029, 0.067]	$+9.3$ [3.6, 14.8]	–	$+31.8$ [2.3, 63.0]
Contrast	embedded: target vs. distractor	5.4 [4.4, 6.6] apart	-0.000 [-0.005, 0.005]	-0.0 [-0.9, 0.8]	$+1.9$ [-1.2, 5.1]	–
	appended: target vs. distractor	3.3 [2.5, 4.4] apart	-0.002 [-0.005, 0.002]	-0.0 [-0.5, 0.4]	-1.1 [-2.6, 0.2]	–
Verified second instance (n=26)	embedded: target vs. distractor	6.4 [4.3, 8.9] apart	$+0.001$ [-0.007, 0.011]	-0.5 [-1.9, 0.7]	$+2.4$ [-4.1, 8.4]	–
	appended: target vs. distractor	4.0 [2.3, 6.8] apart	$+0.003$ [-0.002, 0.009]	-0.1 [-1.0, 0.7]	-1.6 [-4.5, 0.6]	–
Opposite sides of center line (n=31)	embedded: target vs. distractor	5.3 [3.7, 7.4] apart	$+0.002$ [-0.005, 0.008]	-0.4 [-1.4, 0.6]	$+0.2$ [-4.0, 2.9]	–
	appended: target vs. distractor	4.3 [2.7, 6.8] apart	-0.001 [-0.008, 0.004]	$+0.0$ [-0.9, 0.9]	-1.4 [-3.9, 0.4]	–

F Q1 Table and Probe Curves

Table 8 gives the per-view reasoner grounding results summarized in Section 3; Table 9 gives the token probes of Section 4, Figure 5 plots them at every captured layer, and Figure 3 and Figure 4 show the paired design and the pixel edits of Section 4 on the median-effect frame.

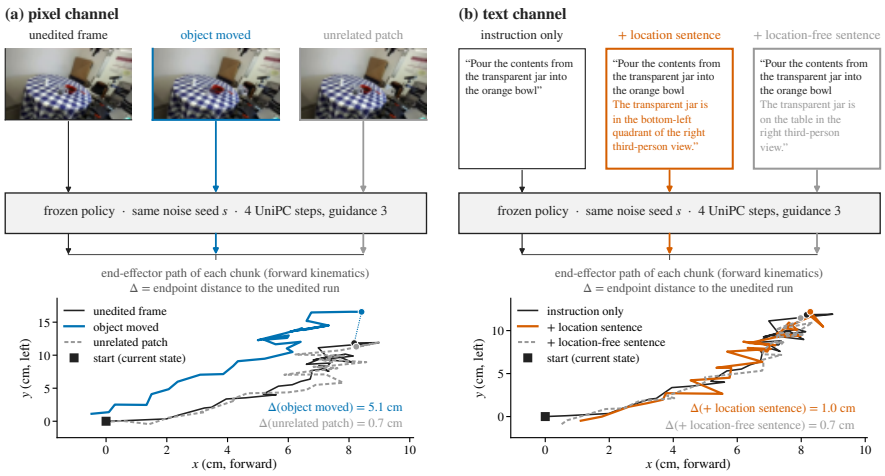


Figure 3: **Paired causal design**, on the median-effect D-DROID frame of Figure 4. The unedited input, an intervention, and a matched control in the same channel (**a**: pixels, Section 4; **b**: text, Section 5) run through the frozen policy under the same noise seed. Forward kinematics gives each chunk's end-effector path; the endpoint shift Δ is the distance to the unedited endpoint. Each test compares the intervention's Δ with the control's over 500 frames (Table 1).

Table 8: **Q1: Reasoner grounding on policy frames** (VLM mode, greedy decoding). Left: the policy checkpoint's own reasoner stream and vision encoder; an answer without a point counts as a miss. Right: the base Cosmos3-Nano, for comparison (it always returns a point). A hit means the predicted point falls inside the ground-truth SAM mask; tolerance allows 5% of the image diagonal; same-class counts a hit on any detection of the target class; box IoU and quadrant use Cosmos3-Nano's box and quadrant answers. Bottom: decomposition of the canvas drop on the 1,001 frames with exterior ground truth, same point prompt, scene-cluster 95% CIs; each input adds one property of the policy's canvas; differences are paired.

		Policy checkpoint's reasoner			Base Cosmos3-Nano		
View	n	Point hit ↑	5% tol. ↑	Same-class ↑	Point hit ↑	IoU>0.5 ↑	Quadrant ↑
Exterior 1	735	0.75 [0.72, 0.79]	0.86	0.89	0.78 [0.75, 0.82]	0.75	0.75
Exterior 2	719	0.76 [0.72, 0.79]	0.85	0.91	0.76 [0.72, 0.80]	0.75	0.75
Wrist	518	0.71 [0.66, 0.75]	0.80	0.83	0.78 [0.74, 0.82]	–	–
Canvas (policy input)	1,169	0.46 [0.43, 0.49]	–	–	0.50 [0.46, 0.53]	–	–
Canvas decomposition (same 1,001 frames)							
Best exterior view, native 360×640	1,001	0.76 [0.72, 0.79]			0.78		
downscaled to 180×320	1,001	0.61 [0.58, 0.65]	resolution: −0.14 [−0.17, −0.11]		0.66	−0.12 [−0.15, −0.09]	
+ second exterior view beside it	1,001	0.44 [0.41, 0.48]	second view: −0.17 [−0.20, −0.14]		0.57	−0.09 [−0.12, −0.05]	
+ wrist view on top (policy canvas)	1,001	0.47 [0.43, 0.50]	wrist view: +0.02 [−0.01, +0.06]		0.50	−0.08 [−0.11, −0.04]	
Three views, exteriors at native resolution	1,001	0.63 [0.60, 0.67]	layout alone: −0.12 [−0.16, −0.09]		0.66	−0.12 [−0.15, −0.09]	

Table 9: **Q2: linear token probes for "this canvas cell overlaps the target"**, evaluated on 176 frames from held-out scenes (95% scene-cluster bootstrap CIs). Policy-reasoner rows: maps from the policy checkpoint's reasoner stream and vision encoder (`src/loop4_w1_reasoner_maps.py`, `src/loop4_probe.py`); the other rows are from the base-checkpoint run (Cosmos3-Nano). *Swap*: the same probe applied to maps recomputed with the target noun replaced by another object in the scene (n=419 frames): mean correlation between a frame's two probe maps, fraction of frames whose top cell changes, and fraction whose top cell lies on the swapped object before → after the swap. MLP: top-cell hit of a 2-layer MLP probe on the same tokens and split.

Source	Layer	AUROC	Top-cell hit	MLP hit	Swap: map corr.	Swap: top changed	Swap: on other obj.
chance / position-only	–	– / 0.69	0.04 / 0.24 [0.17, 0.31]	–	–	–	–
Generator clean canvas	8	0.81	0.27 [0.20, 0.35]	0.23	0.94	0.38	0.05 → 0.07
	20	0.81	0.34 [0.26, 0.42]	0.36	0.90	0.53	0.07 → 0.11
	32	0.80	0.35 [0.28, 0.42]	0.28	0.93	0.48	0.08 → 0.09
Cosmos3-Nano, question after image	8	0.79	0.23 [0.16, 0.31]	–	–	–	–
	20	0.79	0.28 [0.21, 0.36]	–	–	–	–
	28	0.79	0.26 [0.17, 0.35]	–	–	–	–
Policy reasoner, target named before image	8	0.81	0.34 [0.26, 0.41]	–	–	–	–
	12	0.84	0.41 [0.34, 0.48]	–	–	–	–
	20	**0.85**	**0.55** [0.47, 0.63]	–	–	–	–
	28	0.83	0.53 [0.44, 0.61]	–	–	–	–
Cosmos3-Nano, target named before image	8	0.81	0.32 [0.25, 0.40]	–	0.96	0.47	0.09 → 0.17
	12	0.84	0.42 [0.34, 0.50]	0.43	0.89	0.73	0.06 → 0.17
	20	0.86	0.60 [0.52, 0.67]	0.50	0.76	0.91	0.05 → 0.33
	28	0.84	0.52 [0.44, 0.61]	0.47	0.79	0.90	0.04 → 0.31

Table 10: **Peak-based and mask-level probe readouts** (same split and probe as Table 9, refit with one common negative draw for every source, `src/loop4_probe.py`; scene-cluster 95% CIs, which differ slightly from Table 9 because of the refit). Top-cell hit: highest-scoring cell overlaps the mask. Token AUROC: over all test tokens; frame AUROC: mean per frame. Mass in mask: share of the probe's sigmoid mass on mask cells (mask area: 0.04 of the cells). Top-k IoU: overlap of the k highest-scoring cells with the k mask cells.

Source	Top-cell hit	Token AUROC	Frame AUROC	Mass in mask	Top-k IoU
position-only	0.24 [0.17, 0.31]	0.69	0.76	0.05 [0.04, 0.05]	0.12 [0.10, 0.15]
generator clean canvas, L20	0.36 [0.27, 0.45]	0.81	0.86	0.10 [0.09, 0.12]	0.21 [0.17, 0.25]
generator clean canvas, L32	0.37 [0.29, 0.45]	0.81	0.84	0.09 [0.07, 0.10]	0.18 [0.15, 0.21]
Cosmos3-Nano reasoner, L20	0.60 [0.51, 0.67]	0.86	0.89	0.11 [0.10, 0.13]	0.30 [0.26, 0.34]
policy reasoner, L20	0.55 [0.47, 0.63]	0.85	0.89	0.11 [0.09, 0.13]	0.31 [0.27, 0.36]



Figure 4: **Pixel interventions** on one D-DROID exterior view (instruction: "Pour the contents from the transparent jar into the orange bowl"). Titles give this sample's endpoint shift relative to the original. The sample was chosen as the one whose object-moved shift (5.1 cm) is the median over the 500 intervention frames; the population means are 7.7 (moved), 5.7 (removed), 5.3 (copy added), and 3.7 cm (unrelated patch; Table 1).

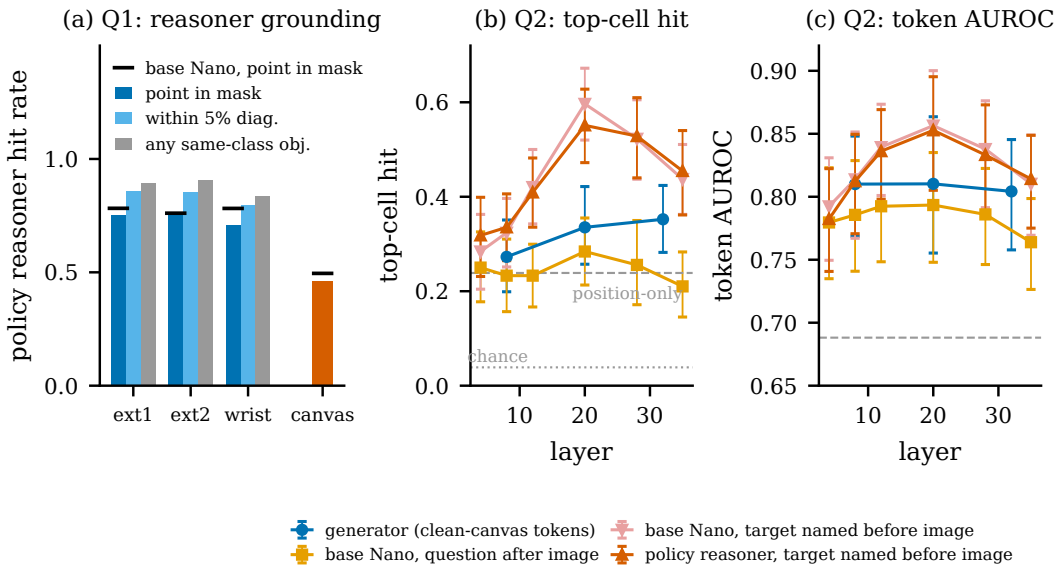


Figure 5: **(a)** Q1: point grounding of the policy checkpoint's reasoner by view (answers without a point count as misses; black ticks: Cosmos3-Nano's point hit; Table 8); on the three-view canvas the policy consumes, hit rate falls to 0.46. **(b, c)** Q2: token-level linear probes for "this canvas token overlaps the target," evaluated on 176 frames from held-out scenes (scene-cluster 95% CIs). Both towers have 36 layers; the generator's clean-canvas tokens (blue) were captured at layers 8, 20, and 32 only, the reasoner's image tokens at layers 4–35 (Table 9 reports four of them): the policy checkpoint's reasoner with the target named before the image (dark red), and Cosmos3-Nano with the target named before the image (light red) or the question asked after it (orange; confounded by causal attention). Dashed line: a probe that sees only the cell index.

Table 11 extends the probes to the generator's other token groups at the last denoising step of the released call. The top block uses states stored mean-pooled per group, so the probe is a ridge regression from the pooled 4,096-d vector to the 17×20 target map. The bottom block uses a second capture of the same call on the same 1,001 frames that keeps every token: the noisy-video maps of frame 1, frame 8, and the mean over frames 1–8 at layers 20 and 32, probed per token with the logistic probe of Table 9, and all 33 action-token states, flattened and probed with ridge regression to the 340-cell map. All are scored by top-cell hit on the same held-out frames.

Table 11: **Probes on the generator's other token groups** (same 825/176 scene split as Table 9; scene-cluster 95% CIs). Pooled: ridge from the mean-pooled group state to the 340-cell target map, best of layers 8–35 (ridge α chosen on held-out training scenes). Per token: logistic probe on each token's state (noisy video) or ridge from the flattened $33\times4{,}096$ action states (ridge α chosen on held-out training scenes), at layers 20 and 32. The position-only prior is 0.24 $[0.17, 0.31]$ and chance 0.04; the policy reasoner's image tokens reach 0.55 $[0.47, 0.63]$ (Cosmos3-Nano: 0.60).

Token group (generator)	Probe	Layer	Top-cell hit
clean canvas (mean-pooled)	ridge, pooled	12	0.29 [0.22, 0.36]
noisy video $f_{1:8}$ (mean-pooled)	ridge, pooled	20	0.27 [0.20, 0.34]
noisy actions $a_{1:32}$ (mean-pooled)	ridge, pooled	24	0.30 [0.22, 0.38]
state token a_0	ridge, pooled	16	0.30 [0.22, 0.38]
last action token a_{32}	ridge, pooled	16	0.26 [0.19, 0.34]
clean canvas (per token, same capture)	logistic, per token	20 / 32	0.35 [0.27, 0.44] / 0.36 [0.28, 0.44]
noisy video, frame 1	logistic, per token	20 / 32	**0.38** [0.29, 0.46] / 0.37 [0.28, 0.46]
noisy video, frame 8	logistic, per token	20 / 32	0.30 [0.22, 0.37] / 0.29 [0.21, 0.38]
noisy video, mean of frames 1–8	logistic, per token	20 / 32	**0.39** [0.29, 0.48] / 0.34 [0.26, 0.42]
all 33 action tokens	ridge, flattened	20 / 32	0.28 [0.21, 0.34] / 0.19 [0.12, 0.25]

G KNOCK-OUT TABLE

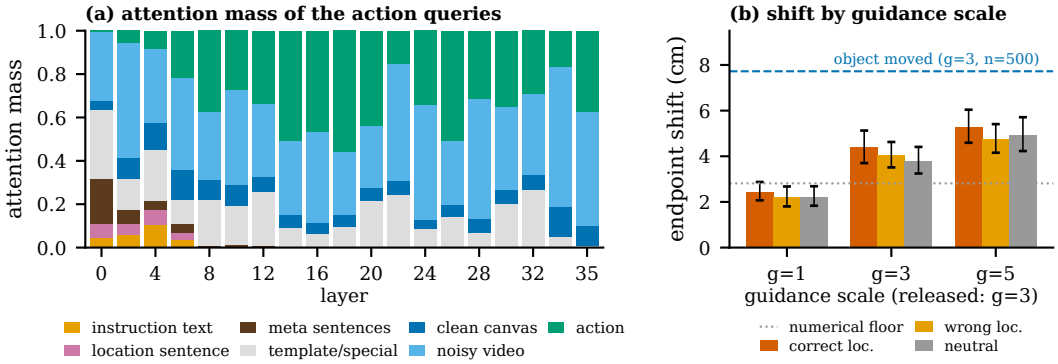


Figure 6: **(a)** Attention mass from the 32 noisy action-token queries to each key group by layer (last denoising step, conditional pass, correct-location prompt, n=200). Content text (instruction: orange; location sentence: purple; meta sentences: brown) receives almost no direct attention from layer 8 on (0.11 of the mass at layer 6), although the knock-out (Table 12) shows it is used through layer 13; grey mass at depth is on template tokens. **(b)** Endpoint shift caused by a location sentence at three guidance scales on the same 200 frames (scene-cluster 95% CIs; Table 5): it scales with guidance, not with whether the sentence is correct, wrong, or location-free. Dashed: pixel object-move effect at the released guidance; dotted: the 2.8 cm floor.

Table 12 gives the full knock-out results discussed in Section 5.

Table 12: **Causal knock-out of text values** (n=200 frames, same seeds). The reference run is the correct-location prompt with kernel compilation off and no knock-out (joint MSE 0.039). Shift: endpoint distance from that run (cm, scene-cluster 95% CI); excess: shift minus the 2.8 cm numerical floor; ΔMSE: paired change in joint MSE to ground truth from the reference, with 95% CI and Wilcoxon p. The prompt-change rows replace the values intervention by a change of the prompt; the neutral-sentence knock-out is the matched control for the location-sentence knock-out (same frames, neutral prompt, its own compile-off reference with MSE 0.039; 13 tokens zeroed vs. 15). "Content text" is the instruction, the location sentence, and the fixed view/duration/resolution sentences.

Intervention	Layers	Shift (cm)	Excess (cm)	ΔMSE	p
kernel compilation off only (numerical floor)	–	2.8 [2.4, 3.2]	0	−0.002	–
neutral sentence appended instead	–	3.4 [3.0, 3.8]	0.6	−0.002	–
location sentence deleted from prompt	–	4.5 [3.9, 5.2]	1.7	−0.002	–
instruction deleted from prompt	–	11.4 [9.8, 13.1]	8.6	+0.008 [0.001, 0.016]	< 0.001
location-sentence values zeroed	all	4.9 [4.3, 5.7]	2.1	−0.0025 [−0.0051, −0.0001]	0.05
neutral-sentence values zeroed (matched control)	all	4.5 [4.0, 5.1]	1.7	−0.0014 [−0.0036, 0.0007]	0.16
instruction values zeroed	all	7.4 [6.3, 8.5]	4.6	+0.002 [−0.004, 0.007]	0.003
all content-text values zeroed	all	16.6 [14.8, 18.4]	13.8	+0.020 [0.012, 0.026]	< 0.001
all content-text values zeroed	4–35	12.0 [10.3, 13.9]	9.2	+0.009 [0.002, 0.016]	0.001
all content-text values zeroed	8–35	11.2 [9.7, 12.9]	8.4	+0.007 [0.000, 0.014]	0.011
all content-text values zeroed	14–35	3.7 [3.2, 4.2]	0.9	−0.000 [−0.002, 0.002]	0.78

H VIT-TOKEN INJECTION

Setup and result. The text channel may be too narrow. As a zero-shot probe that needs no training, we compute the canvas's 340 ViT and merger tokens with the policy checkpoint's own vision tower and inject them at image placeholders in the reasoner stream, with the VAE stream unchanged. The injected tokens are out of distribution: the endpoint moves 12.0 cm from the released prediction and MSE rises from 0.045 to 0.052 (ΔMSE +0.007 [0.003, 0.011]). Moving the object only in the injected view shifts the endpoint by 3.4 cm [3.1, 3.8], against 3.0 cm [2.8, 3.3] for an unrelated patch (Section 4; difference 0.4 cm [0.1, 0.7], moved larger on 56% of samples), with no image-plane pull (0 px [−1.4, +1.2]). A 0.4 cm margin at a 2.8 cm numerical floor, with no direction, is not a usable

signal: the released model has no zero-shot location pathway from image tokens in the reasoner stream to the actions. Whether training could open one is untested.

I Closed-Loop Details: Quadrants, Point Range, Failure Modes

In simulation the reasoner's quadrant answers agree with the object's true quadrant only 53% of the time (267 of 500 first frames) even though its points are accurate. This is a property of the benchmark, not a bug: the pick target is placed near the image center (median $|x-0.5|=0.03$, $|y-0.5|=0.04$ in normalized coordinates; 89% of targets lie within 0.05 of a quadrant boundary), so the quadrant of an accurate point is nearly a coin flip (the reasoner's own point falls in the true quadrant 80% of the time, its verbal quadrant matches its point 60% of the time, and on the 55 episodes with the target ≥ 0.05 from both boundaries its quadrant accuracy is 71%). The "wrong quadrant" condition is therefore correct about as often as the "reasoner quadrant" one; stratifying the quadrant condition by whether its quadrant was correct gives paired differences of $+0.4$ pp (n=267) and -1.3 pp (n=233), with baseline success 59.2% in both strata. The point conditions, which do not have this problem, carry the closed-loop argument; stratifying the reasoner-point condition by whether the appended first point lies within 20 px of the nearest projected point of the target (the reference of Section 6) gives -1.1 pp (n=447) and $+5.7$ pp (n=53) (`src/loop5_final.py`).

Range of the point cue. From the per-step location log of the reasoner-point condition (`src/loop4_w3_closedloop.py`), the reasoner's first point has x between 0.43 and 0.60 and y between 0.39 and 0.57 on 90% of the 500 episodes (5th–95th percentiles), and it drifts by a median 0.016 within an episode. On the 624×352 exterior view the mirrored point lies a median 65 px from the reasoner's (interquartile range 44–87 px) and a median 48 px from the nearest projected point of the target, against 4 px for the reasoner's own point. The mirrored sentence is therefore wrong about the target's position on most episodes, but all point sentences are close to the image center.

Failure modes. Each baseline failure is classified from the stored per-step simulator state: the target's distance to its lifted goal pose (moved if it changed by more than 1 cm), the gripper's touching and holding flags for the target, and the projected gripper and target points in the exterior camera. Of the 204 failures, 102 touch or move the target without holding it, 9 hold it at some replanning step and end without success, 46 never touch it but bring the gripper within 0.05 (normalized image units) of it, and 47 never approach it. Other objects' states are not stored, so a grasp of the wrong object cannot be identified directly; the last class bounds the failures a location cue could repair by redirecting the reach. The class mix barely changes under the location conditions (never approached: 47 released, 50 with the reasoner's point, 50 with the mirrored one), whereas deleting the instruction raises it to 230 of 400 failures. The reasoner-point condition rescues 17% of the 47 never-approached baseline failures and 30% of the contact-without-grasp ones, the mirrored point 15% and 27%: the flips of Section 6 are not concentrated on the failures a location cue could address.

J Does Reasoner Grounding Error Predict Policy Failure?

If the two towers shared a representation of the object, their failures would coincide. On the 1,001 D-DROID frames with ground truth, the policy reasoner's point error on the best exterior view (972 frames with a returned point) predicts whether a sample falls in the policy's worst quartile of endpoint error with AUROC 0.58 $[0.54, 0.62]$ (Spearman 0.15; Cosmos3-Nano: 0.58 $[0.54, 0.62]$ on all 1,001, Spearman 0.16); the generator probe's residual gives 0.52–0.56. The two towers' localization errors are nearly uncorrelated (Spearman 0.04 $[-0.04, 0.10]$; Cosmos3-Nano 0.05 $[-0.02, 0.12]$): they fail on different frames.

K Replication on a Second Checkpoint

To check that the pattern is not peculiar to one set of weights, we repeat the open-loop interventions on Cosmos3-Edge-Policy-DROID, the smaller member of the family: 4B parameters in total by the model card's count (the released weights hold 3.8B), built on a 2B dense 28-layer backbone that is trained from scratch rather than initialized from Qwen3-VL (NVIDIA, 2026), with the same dual-stream design and policy interface, served through the identical released path on the same 500 frames

and seeds (Table 13; joint MSE 0.042, hold-still 0.115). This is a replication of the intervention pattern, not a second instance of the paper's premise: Edge's reasoner stream is not a pretrained VLM, so it has no VLM-mode grounding to compare against, and we did not run its probes, attention analyses, knock-outs, or closed-loop episodes. Within that scope every pixel contrast and every instruction-versus-location contrast reproduces in sign and ordering; Nano's 0.5 cm correct-versus-neutral excess does not (Edge 3.5 against 3.6 cm). The four location sentences shift the endpoint by 3.5–3.6 cm (Nano: 4.0–4.5 cm) with no MSE change (paired ΔMSE within ± 0.001, all CIs containing zero); moving the target in pixels shifts it by 7.7 cm against 3.2, 6.5, and 3.7 cm for the unrelated patch, the artifact-complete control, and a different object moved (target larger by 4.5 $[3.8, 5.3]$, 1.3 $[0.7, 1.8]$, and 4.6 cm $[3.8, 5.5]$), within 0.6 cm of the Nano rows. The positive controls are weaker on Edge: deleting the instruction shifts the endpoint by 13.0 cm against 14.8 on Nano, and swapping the noun by 14.5 cm against 20.8, both still with clearly worse MSE (ΔMSE $+0.013$ and $+0.018$). The smaller model therefore reads the noun less strongly, while its response to location text is if anything smaller and equally indistinguishable from the neutral sentence.

Table 13: **Second checkpoint** (Cosmos3-Edge-Policy-DROID, 4B total, 2B dense backbone trained from scratch). Open-loop interventions on the same 500 frames and seeds as Table 1; endpoint shift in cm with scene-cluster 95% CIs, paired change in joint MSE from the Edge baseline (0.042) with CI; Nano numbers repeated for comparison, and the Nano$-$Edge gap in shift (from unrounded means, so it can differ by 0.1 from the difference of the printed means).

Channel	Condition	Edge shift (cm)	Edge ΔMSE	Nano shift (cm)	Nano ΔMSE	Gap (cm)
Location text	correct location	3.5 [3.2, 3.9]	-0.000 [-0.001, 0.001]	4.5	$+0.001$	1.0
	wrong location	3.6 [3.2, 3.9]	$+0.001$ [-0.000, 0.003]	4.2	$+0.001$	0.7
	reasoner location	3.5 [3.2, 3.9]	$+0.000$ [-0.001, 0.002]	4.5	$+0.002$	1.0
	neutral sentence	3.6 [3.2, 4.0]	-0.000 [-0.002, 0.001]	4.0	-0.000	0.4
Instruction	paraphrase (n=495)	5.2 [4.6, 5.9]	$+0.000$ [-0.002, 0.002]	5.6	$+0.001$	0.4
	instruction deleted	13.0 [11.8, 14.3]	$+0.013$ [0.009, 0.017]	14.8	$+0.016$	1.8
	object swapped (n=419)	14.5 [13.0, 16.1]	$+0.018$ [0.013, 0.023]	20.8	$+0.026$	6.3
Pixels	object moved	7.7 [7.0, 8.5]	–	7.7	–	0.0
	object removed	6.0 [5.4, 6.6]	–	5.7	–	-0.3
	unrelated patch	3.2 [2.9, 3.5]	–	3.7	–	0.6
	patch, object removed	6.5 [5.8, 7.2]	–	6.4	–	0.0
	other object moved (n=367)	3.7 [3.3, 4.1]	–	4.3	–	0.6