SENSORIMOTOR ALIGNMENT: A PLATONIC PROXY FOR EMBODIED POLICY SUCCESS

Anonymous authors
Paper under double-blind review

ABSTRACT

Robotic foundation models are evaluated through costly closed-loop rollouts, motivating cheap offline proxies for model capability. The Platonic Representation Hypothesis (PRH) (Huh et al., 2024) suggests that as representations improve, different modalities such as vision and language converge toward a shared statistical model of reality. We ask whether motion trajectories form another such modality: as robot models become more capable, should their sensory representations converge toward the structure of physical action? To examine this, we propose **Sensorimotor Alignment** (S^2), a training-free offline metric that measures how well a model's sensory representation aligns with the geometry of physical action by comparing sensory and trajectory similarity structures. Crucially, S^2 is consistently and positively related to downstream robot task success across embodied model families. This relation makes S^2 a low-cost offline proxy for identifying which models are likely to achieve higher task success before closed-loop evaluation. It can also identify promising VLM backbones even *before* costly VLA fine-tuning. Moreover, S^2 offers representation-level explanations for phenomena observed in VLAs and WAMs. More broadly, S^2 brings PRH into embodiment by making sensorimotor convergence a measurable object.

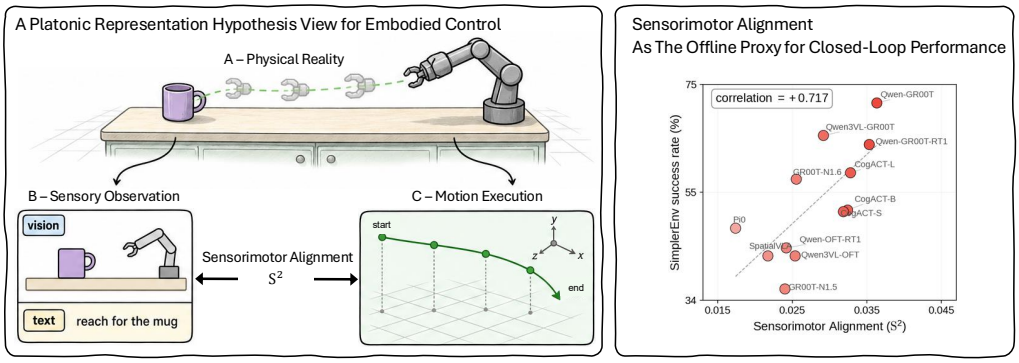


Figure 1: **From the Platonic Representation Hypothesis (PRH) to an offline metric of robot capability.** We view sensory observations and robot motions as two projections of the same physical reality. Under an embodied interpretation of PRH, the similarity structure induced by a capable robot model's visual-language representation should converge with that of physical trajectories. We formalize this property as **Sensorimotor Alignment** (S^2), the alignment between sensory and motor similarity structures over offline robot episodes. Across VLAs, higher S^2 accompanies higher closed-loop success, providing a training-free proxy to prioritize model evaluations.

1 INTRODUCTION

Robotic foundation models increasingly build on large pretrained vision-language models (Beyer et al., 2024; Bai et al., 2025b), video models (Wan Team et al., 2025), and world action models (Feng et al., 2025; Li et al., 2026; Bi et al., 2025). Yet selecting and evaluating these models remains expensive: the most reliable test is still closed-loop rollout in simulation (Li et al., 2024b; Mu et al., 2025) or the real world (Atreya et al., 2025), where evaluation is slow, noisy, and benchmark-dependent. This bottleneck motivates low-cost offline proxies for robotic capability.

A natural approach is to use existing perception-side proxies, such as visual question answering scores for VLMs (Liu et al., 2024; Yue et al., 2024) or pixel-level video quality for world models (Heusel

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.

et al., 2017; Wang et al., 2004; Jayasumana et al., 2024). However, these metrics measure passive perception or generation quality, not whether a representation is organized for action (Zhang et al., 2026a; Kachaev et al., 2026; Tian et al., 2023; Pai et al., 2025). Two scenes may look visually different but require similar physical behavior; conversely, two visually similar scenes may demand different trajectories. For embodied control, the relevant question is therefore not only whether a model understands what is in the scene, but also whether its representation organizes episodes according to the physical behavior each episode requires.

This perspective connects robotics to the Platonic Representation Hypothesis (PRH) (Huh et al., 2024), which argues that as models scale and solve diverse tasks, they are converging to a shared statistical model of reality in their representation spaces. For example, images and text are both projections of a common underlying reality. As vision and language models grow, they measure distances between datapoints increasingly similarly. In this work, we bring PRH to robotics and ask: can physical actions be viewed as another projection of that world, such that scaled embodied models better align vision and language representations with the structure of motion trajectories?

Testing the scaling claim of PRH directly in robotics is currently infeasible, due to the limited scale and diversity of robot interaction data. Instead, we test a concrete operational implication. If embodied capability depends on recovering task-relevant physical structure, then a model's representation should organize sensory episodes according to the trajectories they require. In other words, episodes that are nearby in representation space should also be nearby in motor space.

We formalize this as **Sensorimotor Alignment** (S^2), a training-free metric that measures the alignment between sensory and motor similarity structures (Figure 1). Given a dataset of robot episodes, we construct a sensory kernel from a model's latent features over image and/or language inputs, and a motor kernel from ground-truth trajectories using a pace-invariant trajectory distance (Sakoe & Chiba, 1978). We then score the structural agreement between these kernels using Centered Kernel Nearest-Neighbor Alignment (CKNNA) (Huh et al., 2024). With high S^2, sensory representations group episodes by visual or instruction similarity and required physical behavior.

We first show that S^2 is positively related to downstream robot task success across diverse model families in both simulated and real-world settings. For Vision-Language-Action models (VLAs), S^2 strongly correlates with closed-loop manipulation success, supporting its use as a control-relevant representation metric. For raw Vision-Language Models (VLMs), S^2 computed before any robot fine-tuning correlates with the success of the resulting VLA policies, whereas general visual question answering (VQA) scores do not. For World Action Models (WAMs), S^2 correlates with downstream success more strongly than standard video quality metrics such as Fréchet Inception Distance (FID) (Heusel et al., 2017), Structural Similarity (SSIM) (Wang et al., 2004), and Peak Signal-to-Noise Ratio (PSNR). For real-robot policies, S^2 computed offline correlates with physical-robot success rates under consistent evaluation conditions. These results suggest that sensorimotor structure better reflects embodied success than passive perception or pixel-level generation quality.

We also use S^2 to explain, at the representation level, four phenomena that prior work reports from closed-loop results alone (Section 4). (i) Fine-tuned VLAs often ignore the instruction and act on what the scene suggests; we find that their success tracks the S^2 of vision features but not of text features. (ii) Freezing the vision encoder during VLA fine-tuning lowers success; we find that freezing also blocks the S^2 gain from end-to-end fine-tuning. (iii) WAMs lose little success when iterative video denoising is skipped; we find that S^2 is already high at the earliest denoising step. (iv) The middle blocks of video diffusion transformers, known to matter most for semantic reasoning, also carry the strongest S^2.

Our contributions are threefold:

- We propose a PRH-inspired view of embodiment: successful robot policies should organize sensory representations according to the structure of physical action.
- We introduce S^2, a training-free offline metric that measures kernel alignment between model-induced sensory features and action-induced motor trajectories.
- We show that S^2 positively correlates with downstream robot task success across VLAs, VLMs, and WAMs, and use it to diagnose how different design choices shape control-relevant representations.

More broadly, S^2 provides a representational way to study embodiment by making sensorimotor convergence empirically measurable, a cheap, complementary perspective to closed-loop success.

2 SENSORIMOTOR ALIGNMENT

Preliminary: the Platonic Representation Hypothesis (PRH) (Huh et al., 2024) argues that as models scale and solve diverse tasks, their internal representations converge toward a shared model of reality. Drawing on Plato's Allegory of the Cave, PRH frames different modalities, such as images or text, as distinct projections of the same underlying world. Consequently, models trained on these diverse inputs naturally recover the shared structure that generated them.

While PRH focuses on observational modalities like vision and language, we study its robotics analogue. We treat robot actions, manifested as trajectories and joint torques, as another projection of the physical world, and ask whether successful embodied models organize sensory representations according to this motor structure. We hypothesize that an embodied model's representations should structurally align with this action space, grouping episodes that not only look or sound similar, but require similar physical behavior. We call this property **Sensorimotor Alignment**, or S^2.

We adopt PRH's core mathematical construct, defining representational alignment as the *alignment between similarity structures*, *i.e.*, a metric over kernels. Rather than requiring features to match directly across modalities, it asks whether two distinct representations yield analogous pairwise relations over the same data. We formalize this alignment through three constructs:

- A **representation** $f : \mathcal{X} \to \mathbb{R}^n$ maps an input domain $\mathcal{X}$ to feature vectors. We analyze two projections of the physical world: (1) *sensory representations*, encoding the agent's vision and/or language inputs, and (2) *motor representations*, encoding physical trajectories and joint torques.
- A **kernel** $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ measures pairwise similarity between data points. We construct separate kernels for the sensory and motor representations to capture their respective similarity structures.
- A **kernel-alignment metric** $m : \mathcal{K} \times \mathcal{K} \to \mathbb{R}$ quantifies the isomorphism between two kernels, measuring how closely the sensory similarity structure aligns with the motor similarity structure.

Sensorimotor Alignment operates on fixed offline episodes. We formalize S^2 over a fixed offline dataset $\mathcal{D} = \{e_i\}_{i=1}^N$ such as DROID (Khazatsky et al., 2024). Each episode $e_i = (\mathbf{x}_i, \boldsymbol{\tau}_i)$ pairs a sensory input $\mathbf{x}_i$ (*e.g.*, a camera image and language instruction) with a ground-truth physical trajectory $\boldsymbol{\tau}_i$, collected via human teleoperation. Standard datasets benchmark all representations against identical physical ground truth, isolating scene understanding from compounding closed-loop errors while ensuring scalable, reproducible evaluation. Sampling details are in Appendix A.2.

The sensory kernel captures sensory feature similarity. For each episode, we extract a feature vector $\mathbf{z}_i$ from the target model given sensory input $\mathbf{x}_i$ using only the first frame for video datasets to match standard VLA architectures. The kernel is then the pairwise cosine similarity of these features:

$$\mathbf{K}_{ij}^{\text{sensory}} = \cos(\mathbf{z}_i, \mathbf{z}_j).$$

This formulation is architecture- and layer-agnostic, enabling direct comparisons across different modalities (vision and/or text) and internal network depths (*e.g.*, vision encoder *vs.* the output layer).

The motor kernel captures physical trajectory similarity. For these episodes, we extract the ground-truth trajectory $\boldsymbol{\tau}_i$, *i.e.*, the 6-DoF end-effector pose expressed in the robot base frame (Cartesian position $\in \mathbb{R}^3$ and orientation $\in \mathrm{SO}(3)$), together with the gripper state. Because identical tasks vary in execution speed, we require a pace-invariant metric. We apply Dynamic Time Warping (DTW) (Sakoe & Chiba, 1978) to compute sequence alignment costs:

$$\mathbf{K}_{ij}^{\text{motor}} = \mathrm{DTW}_{\mathrm{SO}(3)}(\boldsymbol{\tau}_i, \boldsymbol{\tau}_j).$$

The local DTW cost is $\mathrm{SO}(3)$-aware, using geodesic distance for rotations and native coordinates for translations and gripper states. This distance matrix defines the motor similarity structure.

Sensorimotor Alignment scores the structural isomorphism. We instantiate the kernel-alignment metric using Centered Kernel Nearest-Neighbor Alignment (CKNNA) from PRH (Huh et al., 2024), which localizes the cross-covariance measurement to the nearest neighbors of each query episode:

$$\mathrm{S}^2 = \mathrm{CKNNA}\big(\mathbf{K}^{\text{sensory}}, \mathbf{K}^{\text{motor}}\big) \ .$$

A high S^2 score indicates a physically grounded sensory representation of vision and/or instruction, where latent proximity strongly correlates with physical trajectory similarity. Figure 2 unpacks this on a single episode, showing the motor-side top-k neighbor set, the sensory-side top-k neighbor set, and the per-episode agreement that S^2 aggregates.

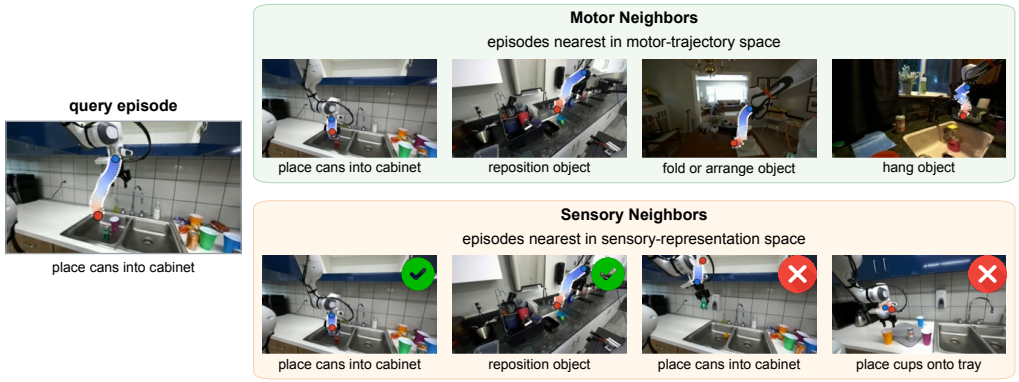


Figure 2: **Sensorimotor Alignment compares neighbors in sensory and motor representation space.** For a query episode, we retrieve its *motor neighbors* by trajectory similarity and its *sensory neighbors* by vision-text representation similarity. Green checks denote sensory neighbors that agree with the motor-neighbor set, while red crosses show visually plausible but motor-inconsistent neighbors. Higher agreement yields a higher S^2 score, indicating that the model's sensory representation reflects the physical similarity structure needed for control.

3 Sensorimotor Alignment Relates to Robot Task Success Rate

The Platonic Representation Hypothesis (PRH) (Huh et al., 2024) posits that as models become more capable, their internal representations converge toward a shared structure of reality. While PRH establishes this for observational modalities like vision and language, robotics provides a harder test case: action is not another passive observation, but a physical trajectory through the world. If embodied capability depends on recovering task-relevant structure, a model's downstream success should be reflected by how well its sensory representations align with the geometry of the action space. In other words, it should group sensory episodes based on the physical behaviors they require.

We test this premise: a model's Sensorimotor Alignment should positively relate to its downstream robot task success. The results carry two implications. First, they provide empirical support for a PRH-inspired view of embodiment: models whose sensory representations better align with motor structure tend to achieve higher task success. Second, S^2 emerges as a training-free evaluation tool: although it cannot replace closed-loop evaluation, it cheaply predicts offline which representations support successful control, even identifying the most capable raw VLM backbones before fine-tuning.

Section 3.1 establishes this positive relation across 3 model families and real-robot policies, Section 3.2 shows that S^2 relates to task success more closely than VQA and video-quality metrics, and Section 3.3 confirms that these findings are robust to experimental design choices.

3.1 S^2 Relates to Task Success Across Model Families

We ask whether stronger alignment between sensory and motor similarity structures accompanies higher task success. In Figure 3, we test this across 3 model families and on policies evaluated on real robots. For each model, we compute S^2 from its sensory features on a reference dataset (x-axis) and compare it with the model's downstream manipulation success rate (y-axis).

Vision-language-action models (VLAs). We test 12 VLAs against their SimplerEnv (Li et al., 2024b) success rates. The correlation $r = +0.717$ is strong, with a model-bootstrap 95% confidence interval of $[+0.436, +0.912]$ (Figure 3a). This supports our premise that S^2 captures a control-relevant representation: models with stronger sensorimotor alignment tend to achieve higher robotics task success. We further check that this correlation does not depend on the reference dataset (BridgeDataV2, Figure 7), model family, model scale, or VLA training setup (Appendix B.2).

Vision-language models (VLMs). We ask whether S^2 computed on a raw VLM, before any robot fine-tuning, relates to the success of the VLA built from it. We compute S^2 for 8 raw VLM backbones from VLM4VLA (Zhang et al., 2026a) on the DROID dataset (Khazatsky et al., 2024) and compare it with the SimplerEnv (Li et al., 2024b) success of the VLAs that VLM4VLA fine-tunes from them. We find a strong correlation with downstream VLA success: $r = +0.681$ with one-tailed $p = 0.031$ (Figure 3b). This demonstrates the practical utility of S^2: models with stronger

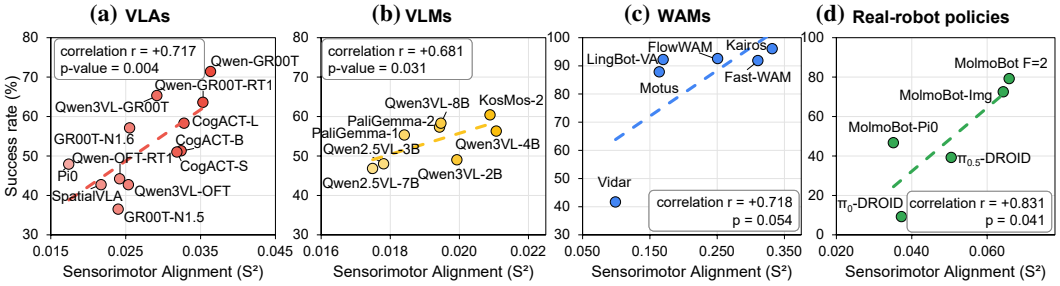


Figure 3: S^2 **correlates with task success across diverse model families and both simulated and real-world task evaluations.** Each panel plots offline S^2 (x-axis) against downstream success (y-axis) for (a) finetuned VLAs, (b) raw VLMs, (c) WAMs, and (d) real-robot policies. These correlations support S^2 as a paradigm-agnostic metric.

sensorimotor alignment before fine-tuning tend to achieve higher downstream task success, so S^2 can identify promising VLM backbones even before costly VLA fine-tuning.

World action models (WAMs). Finally, we ask whether the relation extends to WAMs, which build on video generators rather than VLMs. We compute S^2 for 6 WAMs on the bimanual ALOHA dataset (Zhao et al., 2023) and compare it with their success on RoboTwin 2.0 (Mu et al., 2025). Across these 6 models, S^2 shows a strong correlation with success, $r = +0.718$ with one-tailed $p = 0.054$ (Figure 3c). Appendix A.4 lists the 6 WAM models and shared evaluation protocol.

Real-robot policies. We test whether the relation in simulation between S^2 and task success extends to real robots. Because real-world success depends on the task set, robot, cameras, and protocol, we use the physical-robot success rates that MolmoB0T (Deshpande et al., 2026) reports for 5 policies on the same 40 tasks, and compute S^2 for their public checkpoints on the same 6,000-episode DROID subset. Across these 5 policies, S^2 shows a strong correlation with real-robot task success, $r = +0.831$ with one-tailed $p = 0.041$ (Figure 3d). Appendix A.4 describes the shared evaluation conditions and checkpoint scoring.

We further validate that S^2 is related to downstream task success for frozen vision backbones under low-data behavior cloning and visual-reward learning (Appendix B.1).

3.2 S^2 Relates to Task Success More Closely Than Alternative Metrics

Prior work asks whether offline evaluation scores predict downstream control performance. VLM4VLA (Zhang et al., 2026a) examines the correlation between VQA benchmark scores and the performance of the resulting VLA policies, finding weak or even negative correlations. For video models, Tian et al. (2023) compare the quality of WAM-generated video with downstream control performance and find that the agreement is inconsistent across tasks. A common limitation is that these metrics measure language understanding or pixel-level fidelity, neither of which is grounded in the physical actions the robot must execute. S^2 addresses this gap by measuring whether episodes with similar physical trajectories are also similar in the model's sensory representation, directly linking perception to action. As we show below, S^2 correlates more strongly with task success rate than any VQA or video-quality metric we evaluate.

S^2 **correlates with VLA success where VQA scores do not.** We compare S^2 with aggregate VQA scores as offline proxies of downstream VLA success, using the same 8 raw VLM backbones and evaluation setting as in Section 3.1. We find that VQA scores show virtually no correlation with VLA success ($r = -0.154$; Figure 4, right), whereas S^2 correlates strongly ($r = +0.681$, one-tailed $p = 0.031$; Figure 4, left). S^2 thus tracks VLA success where VQA benchmarks do not.

S^2 **correlates with WAM success more closely than video-quality metrics.** We compare S^2 with standard video-quality metrics (FID (Heusel et al., 2017), SSIM (Wang et al., 2004), and PSNR) on 3 of the WAMs, Motus, LingBot-VA, and Vidar, evaluated on RoboTwin. We find that video-quality metrics show no significant correlations with task success: SSIM reaches only $r = +0.558$ and PSNR only $r = +0.468$ (Figure 5). FID even correlates in the wrong direction. Because lower FID indicates higher video quality, a useful FID proxy would correlate negatively with success; instead, we find $r = +0.680$. In contrast, S^2 computed on ALOHA episodes correlates far more strongly ($r = +0.999$) and is the only metric that correctly recovers the success-rate ordering. Pixel-level forecasting fidelity thus shows little relation to the control success that S^2 captures.

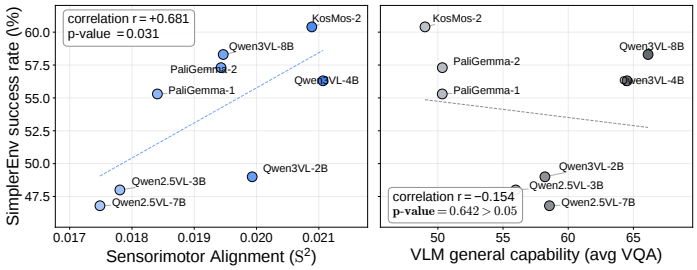


Figure 4: **Raw VLM S^2 correlates with downstream VLA success.** Across 8 models (Zhang et al., 2026a), raw VLM VQA scores (right) show virtually no correlation with finetuned SimplerEnv policy success. S^2 (left) correlates strongly with policy success, offering a cheap guide to model choice.

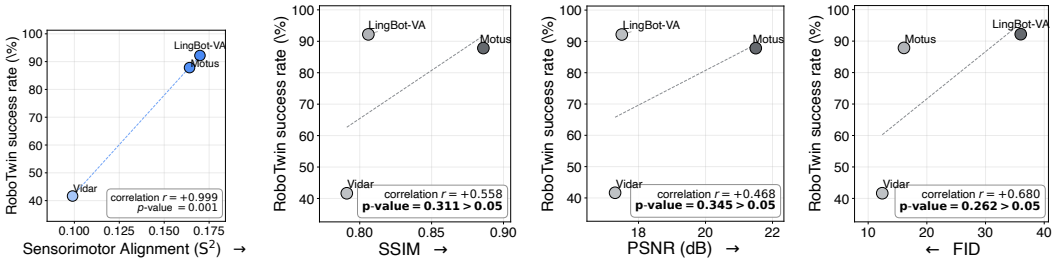


Figure 5: S^2 **correlates with WAM success where perceptual metrics do not.** For 3 WAM policies evaluated on RoboTwin, standard video-quality metrics (SSIM, PSNR, FID) show no significant correlations with task success. In contrast, only S^2 computed on ALOHA episodes correctly recovers the success-rate ordering. Only 3 of the 6 WAMs are compared, as the others do not decode future video under the control-evaluation settings.

3.3 S^2 Is Robust to Evaluation Choices

A useful offline metric should satisfy two complementary properties: stability to nuisance choices in how it is computed and sensitivity to physically meaningful mismatches between the reference data and the downstream task. We vary 9 factors in evaluating S^2 and find that the correlation with success stays positive under all 8 nuisance choices and collapses only when the reference data have the wrong arm configuration. Two nuisance factors are presented below; Appendix C.1 tests the remaining six, Appendix C.3 details the arm-configuration collapse, and Appendix E reports full per-model S^2 values at every horizon.

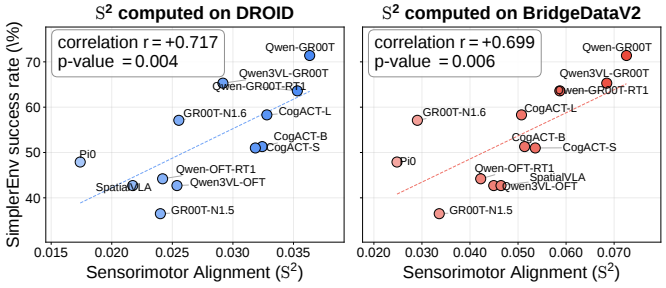


Figure 6: **Cross-dataset generalization.** For 12 VLAs, S^2 on DROID (left) and BridgeDataV2 (right) correlates strongly with SimplerEnv-WidowX success, reflecting capability rather than dataset artifacts.

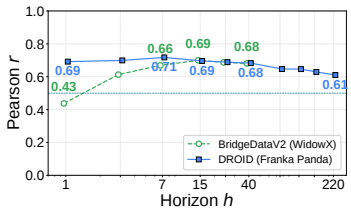


Figure 7: S^2 **correlation is stable across trajectory horizons.** For 12 VLAs, the correlation between S^2 and success remains robust ($r > 0.5$) across nearly all evaluated horizons (h) on BridgeDataV2 and DROID.

The correlation transfers across reference datasets. We validate that S^2 is not tied to a particular reference dataset. We compute S^2 on both DROID (Franka Panda) and BridgeDataV2 (Walke et al., 2023) (WidowX), two datasets with different arm kinematics. Both yield strong correlations near $r = +0.7$ with downstream success for the 12 VLAs on SimplerEnv (Figure 6), confirming that S^2 captures a transferable property of the representation rather than a dataset-specific artifact.

The correlation is stable across trajectory horizons. We validate that S^2 is not sensitive to the trajectory horizon h used during evaluation. Across both reference datasets, the correlation remains above $r = 0.5$ from short horizons through $h = 220$ (Figure 7), confirming that S^2 reflects a persistent structural property rather than an artifact of a particular horizon choice.

4 REPRESENTATION-LEVEL EXPLANATIONS WITH S^2

Prior work reports several phenomena in VLAs and WAMs based only on closed-loop results. Because S^2 relates to task success (Section 3), we use it to provide a representation-level explanation for each of these phenomena, as summarized below:

- **Modality dominance** (Section 4.1): prior work observes that VLAs often struggle with instruction grounding (Fei et al., 2025; Fang et al., 2026; Zhang et al., 2026b). We explain that after VLA fine-tuning, task success is associated with vision-feature S^2 but not text-feature S^2, indicating that fine-tuned VLAs rely mainly on visual rather than language representations for control.
- **Frozen encoders** (Section 4.2): prior work observes that freezing the vision encoder during VLA fine-tuning degrades downstream task performance (Zhang et al., 2026a). We explain that freezing also prevents the S^2 gain from end-to-end fine-tuning, indicating that fine-tuning improves control by aligning the visual representation more closely with motor structure.
- **Iterative denoising** (Section 4.3): prior work observes that WAMs can skip test-time video generation with little loss in task success (Yuan et al., 2026). We explain that S^2 is already high at the earliest denoising step and changes little afterwards, indicating that the sensorimotor structure needed for control is present before denoising finishes.
- **Mid-depth blocks** (Section 4.4): prior work observes that mid-depth blocks of video diffusion transformers are especially important for semantic reasoning (Wang et al., 2026). We explain that the same mid-depth blocks carry the strongest S^2, indicating that the layers that reason about video semantics are also the layers most aligned with robot actions.

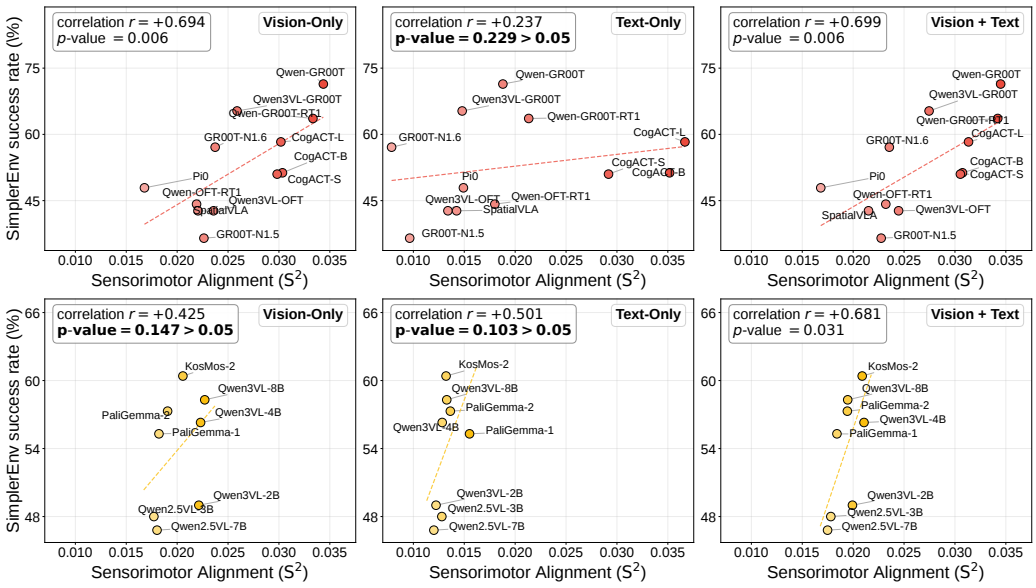


Figure 8: **Fine-tuned VLAs rely mainly on visual rather than language representations for control.** We correlate S^2 computed from vision features, text features, and joint vision-text features with SimplerEnv success. **Top:** For fine-tuned VLAs, S^2 from vision features and from joint features yields *similarly* strong correlations, while S^2 from text features is weak, indicating that fine-tuned VLAs rely mainly on visual rather than language representations for control. **Bottom:** For raw VLMs before any robot fine-tuning, neither vision-only nor text-only S^2 reaches significance, but the joint vision-text representation still predicts downstream success.

4.1 WHAT CARRIES THE CORRELATION: VISION, TEXT, OR BOTH?

Prior work observes that VLAs often struggle with instruction grounding, acting on what the scene suggests rather than on what the instruction asks (Fei et al., 2025; Fang et al., 2026; Zhang et al., 2026b). To examine this at the representation level, we ask which modality carries the correlation with task success: vision, text, or vision and text together. In Figure 8, we compute S^2 separately for each modality. We take the VLM output tokens read by the action head and form *vision features* from the image tokens, *text features* from the instruction tokens, and *joint vision-text features* from all tokens. The full per-model S^2 values across all feature positions are reported in Appendix D.1.

Fine-tuned VLAs rely mainly on visual rather than language representations for control. At horizon $h=3$, we find that S^2 from vision features correlates with the SimplerEnv success of the 12 fine-tuned VLAs nearly as strongly as S^2 from joint vision-text features ($r=+0.694$ and $r=+0.699$), while S^2 from text features shows no significant relation ($r=+0.237$). Before fine-tuning, the pattern differs: in raw VLMs, S^2 from text features ($r=+0.501$) correlates more strongly than S^2 from vision features ($r=+0.425$), and S^2 from joint vision-text features correlates most strongly ($r=+0.681$).

We therefore infer that fine-tuned VLAs rely mainly on visual rather than language representations for control, which explains their weak instruction grounding at the representation level.

4.2 Freezing Vision Encoders Limits S^2 Gain

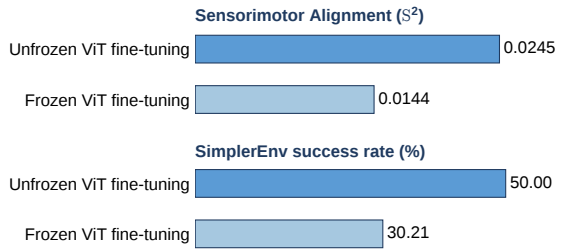
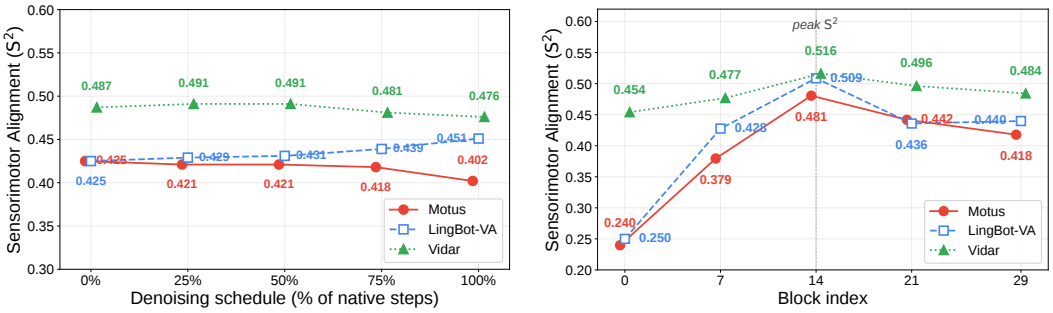


Figure 9: **Freezing the vision encoder ViT reduces both S^2 and downstream success.** For a Qwen2.5-VL-3B VLA fine-tuned on Bridge-Data v2, we compare end-to-end training with a frozen vision encoder. Freezing lowers both S^2 from joint vision-text features (top) and SimplerEnv success (bottom), suggesting that S^2 captures a representation-level bottleneck linked to policy performance. Motion horizon $h=7$.

VLM4VLA (Zhang et al., 2026a) reports that freezing the vision encoder during VLA fine-tuning severely degrades downstream robotics performance. We use S^2 to test whether end-to-end fine-tuning improves control by increasing sensorimotor alignment in the action-head input.

In Figure 9, we compare 2 VLAs differing only in whether the vision encoder is frozen during fine-tuning. For each, we compute S^2 from the joint vision-text features read by the action head and measure SimplerEnv success. Freezing the encoder reduces S^2 by 41% and downstream success by 40%. The S^2 drop holds at all 10 motion horizons, ranging from 38% to 49% (Appendix D.2).

We infer that end-to-end fine-tuning improves control by aligning visual representations with motor structure, explaining why freezing the vision encoder during VLA fine-tuning lowers downstream task success, as reported by VLM4VLA (Zhang et al., 2026a).



(a) S^2 is nearly flat across the schedule.

(b) Block 14 has the strongest S^2 in all 3 WAMs.

Figure 10: **Sensorimotor Alignment emerges early and peaks at mid-depth layers.** (a) S^2 remains nearly flat across the denoising schedule, suggesting that the sensorimotor structure needed for action decoding is present early. (b) S^2 consistently peaks at Block 14, indicating that mid-depth layers carry the strongest sensorimotor signal.

4.3 Early-Denoising Features Already Carry Strong Sensorimotor Alignment

WAMs typically decode actions from video latents obtained through iterative denoising. Fast-WAM (Yuan et al., 2026) shows that this process can be skipped at inference with little loss in task success. We use S^2 to test whether sensorimotor alignment is already present at early denoising steps.

In Figure 10a, we track S^2 along the denoising schedule of 3 WAMs (Bi et al., 2025; Li et al., 2026; Feng et al., 2025) on RoboTwin (Mu et al., 2025). For each model, we extract latent features and compute S^2 at 5 points in its native schedule, from 0% to 100%. S^2 is already high at the earliest point and remains nearly flat, varying by at most 6% from its initial value across all 3 models. We therefore infer that the sensorimotor structure needed for control is present before denoising finishes. Because S^2 correlates with WAM task success (Section 3.1), this offers a representation-level explanation for why WAMs lose little task success when iterative video denoising is skipped at inference.

4.4 Mid-Depth Blocks Carry the Strongest Sensorimotor Alignment

Prior analysis of Chain-of-Steps (Wang et al., 2026) shows that mid-depth layers in Video Diffusion Transformers are especially important for semantic reasoning over objects and interactions. We test whether the strongest sensorimotor signal for embodied control also lies within this semantic window.

In Figure 10b, we compute S^2 at different depths of 3 WAMs (Bi et al., 2025; Li et al., 2026; Feng et al., 2025) on RoboTwin. All 3 share a 30-block video backbone, and we extract features from Blocks 0, 7, 14, 21, and 29. As shown in Figure 10b, Block 14 consistently yields the highest S^2 across all 3 models. We find that this peak falls within the mid-depth semantic window identified by Chain-of-Steps (Wang et al., 2026), and infer that the layers that reason about video semantics are also the layers most aligned with robot actions. This offers a representation-level explanation for how the semantic reasoning that Chain-of-Steps locates in mid-depth video diffusion blocks relates to embodied control. Full analysis in Appendix D.3.

5 Related Work

Representation learning in robotics. Many robotics studies use representation learning to improve control. R3M (Nair et al., 2022) uses a time-contrastive objective and a video-language alignment loss. MVP (Radosavovic et al., 2022) and VC-1 (Majumdar et al., 2023) pretrain a masked autoencoder on egocentric video. Which representational structure explains downstream policy success remains unknown. Hu et al. (2023) show that the ranking of frozen vision backbones changes with the policy-learning method used to evaluate them, so comparing backbones requires training many downstream policies. Burns et al. (2023) compare 15 pretrained vision models and find that emergent segmentation IoU predicts out-of-distribution manipulation better than ImageNet accuracy or shape bias. Theia (Shang et al., 2024) hypothesizes that higher feature-norm entropy improves robot learning. Both predictors are intrinsic statistics of a single vision encoder, ignore the robot's behavior, and were tested only on vision-only backbones. By contrast, S^2 aligns neighborhoods in representation and proprioceptive trajectory spaces to ground evaluation in robot behavior.

Platonic Representation Hypothesis. The Platonic Representation Hypothesis (PRH) (Huh et al., 2024) argues that internal representations of vision and language models converge toward a shared structure of reality, as distinct projections of the same underlying world. Recent work has carried this lens into several new domains. In image generation, REPA (Yu et al., 2025) aligns diffusion-transformer features with self-supervised vision encoders, improving training speed and quality. CREPA (Hwang et al., 2025) extends this recipe to video diffusion. In multimodal understanding, VIRAL (Yoon et al., 2025) re-aligns multimodal LLMs with their vision encoders, recovering lost structure. However, PRH has not yet been brought into robotics. We extend PRH to embodied control, treating robot trajectories as another projection of the world described by vision and language.

6 Conclusion

We bring a PRH-inspired view to embodied control by treating physical trajectories as an additional modality, and introduce Sensorimotor Alignment (S^2), a training-free metric for measuring alignment between sensory representations and physical actions. Across VLAs, raw VLMs, and WAMs, S^2 is consistently and positively related to downstream task success and offers representation-level explanations for how modality, fine-tuning, denoising, and depth affect control. While we use S^2 only to prioritize models offline and offer representation-level explanations, its strong correlation with downstream success suggests optimizing embodied models directly for sensorimotor alignment.

Limitations. The positive relationship between S^2 and task success is correlational, not causal: we show that higher alignment accompanies higher success across the tested models, but we do not demonstrate that improving S^2 will produce better control. Models with similar S^2 values can still achieve different success rates, so S^2 serves as a cheap offline proxy for identifying promising backbones before costly fine-tuning or rollout evaluation, but cannot replace closed-loop testing for final model selection. Our evaluation covers limited model families, benchmarks, and training recipes; broader generalization remains to be tested.

Artificial intelligence use statement

We used AI coding assistants to help implement the S^2 computation and evaluation scripts, check experiment records and reported statistics, interpret results, and review mathematical exposition. We also used generative AI tools to help revise the writing, format LaTeX, and check spelling and grammar. We did not use generative AI tools to generate synthetic datasets. We have reviewed the AI-assisted outputs and take responsibility for the final content of this work.

Ethics statement

This work does not involve human subjects or new data collection. All experiments use publicly available robot datasets, models, and policy checkpoints. We ran no physical-robot experiments; real-robot success rates are taken from published evaluations (Deshpande et al., 2026). S^2 is an offline signal and does not replace closed-loop safety and robustness testing before a robot policy is deployed. All authors have read and adhere to the ICLR Code of Ethics.

Reproducibility statement

The definition of S^2, including the sensory kernel, the DTW-based motor kernel, and CKNNA, is given in Section 2. The reference datasets, episode and anchor sampling, and trajectory frame are described in Sections A.1 to A.3. The evaluated models, the sources of their success rates, and the WAM and real-robot evaluation protocols are described in Section A.4. Default settings and alternative implementation choices are reported in Table 2, and per-model S^2 values are given in Sections D.1 and E. All models are public checkpoints except the 2 VLAs in Section 4.2, which we train with the public VLM4VLA recipe (Zhang et al., 2026a). We will release our implementation of S^2 upon acceptance.

References

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 *CoRL*, 2025.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-VL technical report. *arXiv:2511.21631*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report. *arXiv:2502.13923*, 2025b.

Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, et al. PaliGemma: A versatile 3B VLM for transfer. *arXiv:2407.07726*, 2024.

Hongzhe Bi, Hengkai Tan, Shenghao Xie, Zeyuan Wang, Shuhe Huang, Haitian Liu, Ruowen Zhao, Yao Feng, Chendong Xiang, Yinze Rong, et al. Motus: A unified latent action world model. *arXiv:2512.13030*, 2025.

Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, et al. GR00T N1: An open foundation model for generalist humanoid robots. *arXiv:2503.14734*, 2025.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A vision-language-action flow model for general robot control. In *RSS*, 2025.

Kaylee Burns, Zach Witzel, Jubayer Ibn Hamid, Tianhe Yu, Chelsea Finn, and Karol Hausman. What makes pre-trained visual representations successful for robust manipulation? In *CoRL*, 2023.

Yixiang Chen, Peiyan Li, Yuan Xu, Qisen Ma, et al. FlowWAM: Optical flow as a unified action representation for world action models. *arXiv:2607.13017*, 2026.

Abhay Deshpande, Maya Guru, Rose Hendrix, Snehal Jauhri, et al. MolmoB0T: Large-scale simulation enables zero-shot manipulation. *arXiv:2603.16861*, 2026.

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

Senyu Fei, Siyin Wang, Junhao Shi, Zihao Dai, Jikun Cai, Pengfang Qian, Li Ji, Xinzhe He, Shiduo Zhang, Zhaoye Fei, et al. LIBERO-Plus: In-depth robustness analysis of vision-language-action models. *arXiv:2510.13626*, 2025.

Yao Feng, Hengkai Tan, Xinyi Mao, Chendong Xiang, Guodong Liu, Shuhe Huang, Hang Su, and Jun Zhu. Vidar: Embodied video diffusion model for generalist manipulation. *arXiv:2507.12898*, 2025.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *NeurIPS*, 2017.

Yingdong Hu, Renhao Wang, Li Erran Li, and Yang Gao. For pre-trained vision models in motor control, not all policy learning methods are created equal. In *ICML*, 2023.

Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. The Platonic representation hypothesis. In *ICML*, 2024.

Sungwon Hwang, Hyojin Jang, Kinam Kim, Minho Park, and Jaegul Choo. Cross-frame representation alignment for fine-tuning video diffusion models. *arXiv:2506.09229*, 2025.

Sadeep Jayasumana, Srikumar Ramalingam, Andreas Veit, Daniel Glasner, Ayan Chakrabarti, and Sanjiv Kumar. Rethinking FID: Towards a better evaluation metric for image generation. In *CVPR*, 2024.

Nikita Kachaev, Mikhail Kolosov, Daniil Zelezetsky, Alexey K Kovalev, and Aleksandr I Panov. Don't blind your VLA: Aligning visual representations for OOD generalization. In *AAMAS*, 2026.

Kairos Team, Fei Wang, Shan You, Qiming Zhang, et al. Kairos: A regret-aware native world-action model stack for physical AI. *arXiv:2606.16533*, 2026.

Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. DROID: A large-scale in-the-wild robot manipulation dataset. In *RSS*, 2024.

Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Fei Han, Mingrui Yu, Zelin Gao, Nan Xue, Xing Zhu, et al. Causal world modeling for robot control. *arXiv:2601.21998*, 2026.

Qixiu Li, Yaobo Liang, Zeyu Wang, Lin Luo, Xi Chen, Mozheng Liao, Fangyun Wei, Yu Deng, Sicheng Xu, Yizhong Zhang, et al. CogACT: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation. *arXiv:2411.19650*, 2024a.

Xuanlin Li, Kyle Hsu, Jiayuan Gu, Karl Pertsch, Oier Mees, Homer Rich Walke, Chuyuan Fu, Ishikaa Lunawat, Isabel Sieh, Sean Kirmani, et al. Evaluating real-world robot manipulation policies in simulation. In *CoRL*, 2024b.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. MMBench: Is your multi-modal model an all-around player? In *ECCV*, 2024.

Arjun Majumdar, Karmesh Yadav, Sergio Arnaud, Jason Ma, Claire Chen, Sneha Silwal, Aryan Jain, Vincent-Pierre Berges, Tingfan Wu, Jay Vakil, et al. Where are we in the search for an artificial visual cortex for embodied intelligence? In *NeurIPS*, 2023.

Yao Mu, Tianxing Chen, Zanxin Chen, Shijia Peng, Zhiqian Lan, Zeyu Gao, Zhixuan Liang, Qiaojun Yu, Yude Zou, Mingkun Xu, et al. RoboTwin: Dual-arm robot benchmark with generative digital twins. In *CVPR*, 2025.

Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3M: A universal visual representation for robot manipulation. In *CoRL*, 2022.

Jonas Pai, Liam Achenbach, Victoriano Montesinos, Benedek Forrai, Oier Mees, and Elvis Nava. mimic-video: Video-action models for generalizable robot control beyond VLAs. *arXiv:2512.15692*, 2025.

Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, Qixiang Ye, and Furu Wei. KOSMOS-2: Grounding multimodal large language models to the world. In *ICLR*, 2024.

Delin Qu, Haoming Song, Qizhi Chen, Yuanqi Yao, Xinyi Ye, Yan Ding, Zhigang Wang, JiaYuan Gu, Bin Zhao, Dong Wang, et al. SpatialVLA: Exploring spatial representations for visual-language-action model. In *RSS*, 2025.

Ilija Radosavovic, Tete Xiao, Stephen James, Pieter Abbeel, Jitendra Malik, and Trevor Darrell. Real-world robot learning with masked visual pre-training. In *CoRL*, 2022.

H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE TASSP*, 1978.

Jinghuan Shang, Karl Schmeckpeper, Brandon B. May, Maria Vittoria Minniti, Tarik Kelestemur, David Watkins, and Laura Herlant. Theia: Distilling diverse vision foundation models for robot learning. In *CoRL*, 2024.

StarVLA Community. StarVLA: A Lego-like codebase for vision-language-action model developing. *arXiv:2604.05014*, 2026.

Andreas Steiner, André Susano Pinto, Michael Tschannen, Daniel Keysers, Xiao Wang, Yonatan Bitton, Alexey Gritsenko, Matthias Minderer, Anthony Sherbondy, Shangbang Long, et al. PaliGemma 2: A family of versatile VLMs for transfer. *arXiv:2412.03555*, 2024.

Stephen Tian, Chelsea Finn, and Jiajun Wu. A control-centric benchmark for video prediction. In *ICLR*, 2023.

Homer Rich Walke, Kevin Black, Abraham Lee, Moo Jin Kim, Max Du, Chongyi Zheng, Tony Z Zhao, Philippe Hansen-Estruch, Quan Vuong, Andre Wang He, et al. BridgeData V2: A dataset for robot learning at scale. In *CoRL*, 2023.

Wan Team, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv:2503.20314*, 2025.

Ruisi Wang, Zhongang Cai, Fanyi Pu, Junxiang Xu, Wanqi Yin, Maijunxian Wang, Ran Ji, Chenyang Gu, Bo Li, Ziqi Huang, et al. Demystifying video reasoning. *arXiv:2603.16870*, 2026.

Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE TIP*, 2004.

Heeji Yoon, Jaewoo Jung, Junwan Kim, Hyungyu Choi, Heeseong Shin, Sangbeom Lim, Honggyu An, Chaehyun Kim, Jisang Han, Donghyun Kim, et al. Visual representation alignment for multimodal large language models. *arXiv:2509.07979*, 2025.

Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. In *ICLR*, 2025.

Tianyuan Yuan, Zibin Dong, Yicheng Liu, and Hang Zhao. Fast-WAM: Do world action models need test-time future imagination? *arXiv:2603.16666*, 2026.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. In *CVPR*, 2024.

Jianke Zhang, Xiaoyu Chen, Yanjiang Guo, Yucheng Hu, and Jianyu Chen. VLM4VLA: Revisiting vision-language-models in vision-language-action models. In *ICLR*, 2026a.

Ninghao Zhang, Bin Zhu, Shijie Zhou, and Jingjing Chen. Restoring linguistic grounding in VLA models via train-free attention recalibration. *arXiv:2603.06001*, 2026b.

Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *RSS*, 2023.

APPENDIX

Contents

A. Experimental Details . . . 15
A.1. Datasets . . . 15
A.2. Anchor Sampling Across Horizons . . . 15
A.3. Trajectory Reference Frame . . . 15
A.4. Evaluated Models . . . 15
B. Additional Results on S^2 and Task Success . . . 17
B.1. Frozen Vision Encoders Under Downstream Policy Learning . . . 17
B.2. Controlling for Potential Confounds . . . 17
C. Robustness of S^2 . . . 19
C.1. Implementation Choices . . . 19
C.2. DTW Alignment . . . 20
C.3. Embodiment Mismatch Between Reference and Target . . . 20
D. Details of the Representation-Level Explanations . . . 21
D.1. S^2 Across Feature Positions . . . 21
D.2. Frozen Versus Unfrozen Vision Encoder . . . 21
D.3. WAM Features Along the Denoising Schedule . . . 22
E. Full S^2–Success Results Across Horizons . . . 24

A Experimental Details

This section describes datasets, anchor sampling, trajectory reference frames, and evaluated models.

A.1 Datasets

Both datasets contain teleoperated robot demonstrations with RGB images, proprioceptive states, and language task annotations. We represent every state as a 7D vector $s = [x,\ y,\ z,\ \phi,\ \theta,\ \psi,\ g]$, where (x, y, z) is the end-effector position in meters, (ϕ, θ, ψ) are the roll, pitch, and yaw angles in radians, and g is the gripper opening, with 0 closed and 1 open. This shared format lets us apply the same DTW distance and S^2 computation to both datasets.

BridgeDataV2. BridgeDataV2 (Walke et al., 2023) contains 53,192 episodes on a WidowX arm recorded at 5 Hz, spanning 54 tabletop tasks in one laboratory. The raw state has 8 dimensions; we drop the zero-padded channel at index 6. We use the 38,307 episodes with non-empty task labels.

DROID. DROID (Khazatsky et al., 2024) contains 95,658 episodes on a Franka Panda arm recorded at 15 Hz, spanning 86 task categories across 564 scenes in offices, kitchens, and homes, collected by 50 data collectors on 3 continents. We form the 7D state by concatenating the 6D Cartesian end-effector pose with the gripper position. We keep the 78% of episodes that have a language annotation. As the observation image, we use the `exterior_image_1_left` camera at 180×320, the fixed exterior view closest to the third-person camera of BridgeDataV2.

A.2 Anchor Sampling Across Horizons

We compute S^2 at several trajectory horizons h. To keep these evaluations comparable, we draw one sample set at a maximum horizon $H_{\max}$ and reuse it at every shorter horizon. For each retained episode of length L, we draw one anchor frame t uniformly from $[0,\ L-1-H_{\max}]$, so the slice $\boldsymbol{\tau}_{t:t+H_{\max}}$ stays inside the episode. A shorter horizon reads the prefix $\boldsymbol{\tau}_{t:t+h}$ of the same slice, and the image features at frame t are computed once. The horizon sweep therefore changes only the trajectory length, not the episodes, anchors, or sensory inputs.

On BridgeDataV2, we set $H_{\max} = 40$, keep $N = 7{,}762$ episodes with $L > 40$, and sweep $h \in \{1, 3, 7, 15, 25, 40\}$. On DROID, we set $H_{\max} = 220$, which covers about the same fraction of the mean episode. We also require $L \geq 2H_{\max} + 1 = 441$ so that anchors are not concentrated at the start of short episodes. We keep $N = 6{,}000$ episodes and sweep $h \in \{1, 3, 7, 15, 25, 40, 75, 110, 150, 220\}$.

Drawing a new sample set at each horizon would instead change the episodes and anchor positions from one horizon to the next, since longer horizons exclude short episodes.

A.3 Trajectory Reference Frame

We express all end-effector trajectories in each robot's recorded base frame. Robot controllers and policies output actions in this frame, so two episodes that are close under base-frame DTW call for similar action sequences. The base frame is also the only frame available across our data. DROID provides no world frame, and camera frames shift when cameras move between sessions.

A.4 Evaluated Models

Finetuned VLAs. We evaluate 12 VLAs finetuned on BridgeDataV2 (Walke et al., 2023). They are 5 StarVLA (StarVLA Community, 2026) variants (Qwen2.5-GR00T-Bridge, Qwen2.5-GR00T-RT-1-Bridge, Qwen2.5-OFT-RT-1-Bridge, Qwen3-GR00T-RT-1-Bridge, and Qwen3-OFT-RT-1-Bridge), CogACT (Li et al., 2024a) in Small, Base, and Large sizes, GR00T-N1.5-Bridge, GR00T-N1.6-Bridge (Bjorck et al., 2025), Pi0-Bridge (Black et al., 2025), and SpatialVLA-Bridge (Qu et al., 2025). They cover 4 action-head types (flow matching, diffusion, autoregressive, and deterministic MLP) and 3 VLM training regimes, from frozen to end-to-end. SimplerEnv success rates come from the public evaluation code at `https://github.com/starVLA/starVLA` and `https://github.com/DelinQu/SimplerEnv-OpenVLA`.

Raw VLMs. We evaluate the 8 raw VLMs from VLM4VLA (Zhang et al., 2026a), which are KosMos-2 (Peng et al., 2024), PaliGemma (Beyer et al., 2024), PaliGemma 2 (Steiner et al., 2024), Qwen2.5-VL-3B and Qwen2.5-VL-7B (Bai et al., 2025b), and Qwen3-VL-2B, Qwen3-VL-4B, and

Qwen3-VL-8B (Bai et al., 2025a). None of these 20 models was trained on DROID. The finetuned VLAs were trained only on BridgeDataV2, so DROID tests transfer across robots and scenes.

World action models. We evaluate 6 WAMs: Vidar (Feng et al., 2025), LingBot-VA (Li et al., 2026), Motus (Bi et al., 2025), Fast-WAM (Yuan et al., 2026), FlowWAM (Chen et al., 2026), and Kairos (Kairos Team et al., 2026). For all 6 models, we compute S^2 on identical bimanual ALOHA trajectories, fixing feature position, trajectory horizon, neighborhood size, and DTW computation. For each WAM, the sensory representation is the video-generator latent used for downstream action decoding.

Real-robot policies. We use success rates from MolmoB0T's physical-robot evaluation (Deshpande et al., 2026), which tests all policies on the same tasks, robots, and cameras. Policies run on 3 Franka DROID platforms in 4 environments (kitchen, workroom, bedroom, and office) at 2 institutions. Each environment has 10 pick-and-place tasks with 3 trials per task, giving 120 trials per policy. A trial succeeds if the policy places the named object stably in or on the given receptacle within 900 steps. The all-task evaluation compares 5 policies, three MolmoB0T variants (MolmoB0T, MolmoB0T-Img, and MolmoBot-Pi0) and two baselines (π_0-DROID and $\pi_{0.5}$-DROID). We download the exact public checkpoints and compute S^2 on the same 6,000-episode DROID subset with unchanged settings.

B Additional Results on S^2 and Task Success

This section extends Section 3.1 with two analyses. The first tests frozen vision encoders under a shared policy-learning setup. The second checks whether a confound explains the VLA correlation.

B.1 Frozen Vision Encoders Under Downstream Policy Learning

We ask whether S^2 can rank frozen vision backbones without training any policy. Hu et al. (2023) provide a suitable benchmark: they evaluate 14 pretrained vision backbones (9 ResNet-50 and 5 ViT-B/16) on 21 tasks in Meta-World, Robosuite, and Franka-Kitchen, freezing each backbone and training policies on top of it, so only the backbone changes. We compute S^2 for their released checkpoints on the same 6,000-episode DROID subset as in the main paper, with all other settings unchanged. We take the interquartile mean (IQM) success rates directly from their paper for two policy-learning methods. Behavior cloning (BC) learns from expert demonstrations, and we use the smallest budget of 1 demonstration per task (5 on Robosuite). The visual reward function (VRF) method uses reinforcement learning, with rewards computed from distances between the backbone's embeddings of agent and expert observations.

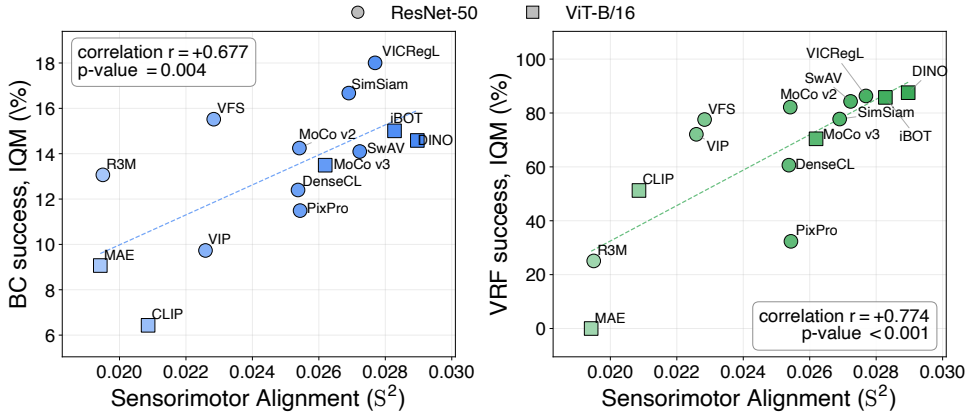


Figure 11: S^2 **of frozen vision encoders correlates with downstream policy success.** Points represent the 14 frozen backbones of Hu et al. (2023), scored by S^2 on DROID without policy training. Success is IQM for behavior cloning at the smallest demonstration budget (left) and visual reward functions over all 21 tasks (right).

We find a strong correlation between the S^2 of the vision backbone and downstream success under both policy-learning methods (Figure 11). For BC, $r = +0.677$ with one-tailed $p = 0.004$. For VRF, $r = +0.774$ with one-tailed $p < 0.001$. The BC correlation becomes weaker as the number of demonstrations grows, dropping to $r = +0.236$ at the largest budget. More demonstrations may let the policy compensate for a weaker representation. For reinforcement learning, we find no positive correlation on two Meta-World seed sets or on Robosuite (r between -0.193 and -0.002). This agrees with Hu et al. (2023), who report that reinforcement-learning results vary widely across random seeds. Overall, S^2 also works for frozen vision backbones. It is most useful when the policy relies directly on the representation, as in low-data imitation or embedding-based rewards.

B.2 Controlling for Potential Confounds

The 12 VLAs in Section 3.1 differ in architecture, model scale, pretraining data, action head, and training recipe at the same time. We run three checks to see whether one of these factors explains the correlation. All checks use the same DROID subset and S^2 settings as the main paper.

Training recipe. The raw-VLM experiment in Section 3.1 holds the recipe fixed. All 8 VLMs are turned into VLAs with the same VLM4VLA (Zhang et al., 2026a) recipe, with the same action head, objective, BridgeDataV2 data, and hyperparameters. The correlation remains strong ($r = +0.681$, one-tailed $p = 0.031$), so it does not depend on differences in training recipe.

Model family and scale. CogACT (Li et al., 2024a) Small, Base, and Large share the same VLM backbone and action-head design and differ only in action-head size. Within this family, S^2 tracks

success closely ($r = +0.826$, one-tailed $p = 0.191$). With only 3 models the test has little statistical power, but the direction and size match the pooled result.

Architecture groups. We remove each of the 5 architecture groups in turn and recompute the correlation on the remaining models (Table 1). The correlation stays positive in all 5 cases, from $r = +0.526$ to $r = +0.785$, so no single group drives the result.

Table 1: **Leave-one-out over architecture groups** for the 12 finetuned VLAs.

Removed group	Pearson's r	One-tailed p
StarVLA	$+0.526$	0.060
CogACT	$+0.785$	0.006
GR00T	$+0.745$	0.007
Pi0	$+0.783$	0.002
SpatialVLA	$+0.687$	0.007

C ROBUSTNESS OF S^2

This section extends Section 3.3. We test six implementation choices, the use of DTW, and a mismatch in arm configuration between the reference data and the target task.

C.1 IMPLEMENTATION CHOICES

We change one choice at a time and keep all other settings at the main-paper defaults, which are the 6,000-episode DROID subset, $h=7$, $k=10$, the LLM-out vision-text feature, and the 12 finetuned VLAs. Table 2 reports the Pearson r between S^2 and SimplerEnv success for each setting. All settings of k, normalization, component weights, rotation distance, and pooling keep the correlation positive and significant, with r between $+0.634$ and $+0.752$.

Table 2: **Sensitivity of the S^2–success correlation to implementation choices.** Pearson's r and one-tailed p across 12 finetuned VLAs. Each block varies one choice; * marks the default.

Setting	r	p
Neighborhood size k		
$k=2$	$+0.679$	0.0076
$k=5$	$+0.667$	0.0089
$k=10^*$	$+0.717$	0.0043
$k=15$	$+0.665$	0.0091
$k=20$	$+0.644$	0.0119
Trajectory normalization		
Absolute pose*	$+0.717$	0.0043
Anchor-relative	$+0.752$	0.0024
Rotation distance		
$\mathrm{SO}(3)$ geodesic*	$+0.717$	0.0043
Quaternion squared chordal	$+0.686$	0.0069
XYZ Euler L_2	$+0.660$	0.0098
Token pooling		
Masked mean*	$+0.717$	0.0043
Masked max	$+0.653$	0.0107
Masked min	$+0.730$	0.0035

Setting	r	p
Component weights (w_p, w_r, w_g)		
$(1,1,1)^*$	$+0.717$	0.0043
$(0.5,1,1)$	$+0.698$	0.0058
$(2,1,1)$	$+0.708$	0.0050
$(1,0.5,1)$	$+0.694$	0.0062
$(1,2,1)$	$+0.703$	0.0054
$(1,1,0.5)$	$+0.702$	0.0055
$(1,1,2)$	$+0.699$	0.0057
$(0,1,1)$	$+0.745$	0.0027
$(1,0,1)$	$+0.658$	0.0100
$(1,1,0)$	$+0.634$	0.0135
Feature position		
Vision encoder output	$+0.567$	0.0274
LLM-in vision-text	$+0.580$	0.0239
LLM-out vision-only	$+0.701$	0.0056
LLM-out vision-text*	$+0.717$	0.0043
LLM-in text-only	-0.325	0.8486
LLM-out text-only	$+0.245$	0.2215

Neighborhood size k. S^2 builds a k-nearest-neighbor graph on each side, and the main paper uses $k=10$ following Huh et al. (2024). Across $k \in \{2, 5, 10, 15, 20\}$, r ranges from $+0.644$ to $+0.717$, with a mild peak at $k=10$.

Trajectory normalization. The anchor-relative setting replaces each position with its offset from the starting frame and keeps rotation and gripper state unchanged. It gives a slightly higher r than absolute poses, possibly because removing the workspace offset emphasizes the shape of the motion. We keep absolute poses for simplicity.

Component weights. The DTW step cost $c = w_p\, d_{\mathrm{pos}} + w_r\, d_{\mathrm{rot}} + w_g\, d_{\mathrm{grip}}$ uses equal weights by default. Halving, doubling, or removing any single weight keeps r between $+0.634$ and $+0.745$. Removing rotation or gripper lowers r most, suggesting both carry information beyond position.

Rotation distance and token pooling. The $\mathrm{SO}(3)$ geodesic gives the highest r of the three rotation distances. For pooling the LLM-output tokens, masked min gives a slightly higher r than the default masked mean, and masked max gives the lowest.

Feature position. We also compute S^2 at earlier feature positions of the 12 VLAs, from the vision-encoder output to the LLM output, and from text features alone. We find that the correlation holds at every visual feature position and grows with depth, from $r=+0.567$ at the vision-encoder output to $r=+0.717$ at the LLM output. Text-only features show no significant relation at either position. This agrees with reports that VLAs follow visual cues even when the language instruction says otherwise (Fang et al., 2026; Zhang et al., 2026b).

C.2 DTW ALIGNMENT

S^2 aligns trajectory pairs with DTW before measuring their distance, so that the same motion performed at different speeds counts as similar. To test whether the correlation depends on this warping, we compare DTW with a lockstep distance that matches frames by time index (Table 3). DTW raises S^2 for all 12 VLAs from $h=7$ onward. The gain grows with the horizon, from 1.2% at $h=7$ to 34.1% at $h=220$, since longer trajectories contain more variation in speed. We find that the correlation with success barely changes across the 10 horizons, from $r=+0.617$–$+0.713$ without DTW to $r=+0.610$–$+0.717$ with DTW, a relative difference between -3.2% and $+0.6\%$. DTW lifts all models by a similar amount ($34 \pm 4\%$ at $h=220$), which leaves their ranking and therefore r nearly unchanged. The S^2–success correlation is therefore not an artifact of the warping.

Table 3: **Comparing S^2 with and without DTW warping of the trajectories.** DTW raises absolute S^2 more at longer horizons h, while success-rate Pearson r stays stable.

	S^2 (mean over 12 VLAs)			S^2–SR Pearson r		
Horizon h	w/ DTW	w/o DTW	rel. Δ	w/ DTW	w/o DTW	rel. Δ
1	0.0268	0.0268	$+0.0\%$	$+0.692$	$+0.692$	$+0.0\%$
3	0.0268	0.0269	-0.6%	$+0.699$	$+0.700$	-0.1%
7	0.0280	0.0277	$+1.2\%$	$+0.717$	$+0.713$	$+0.6\%$
15	0.0314	0.0302	$+4.1\%$	$+0.696$	$+0.710$	-2.0%
25	0.0342	0.0317	$+7.7\%$	$+0.689$	$+0.711$	-3.2%
40	0.0410	0.0365	$+12.4\%$	$+0.683$	$+0.680$	$+0.4\%$
75	0.0537	0.0442	$+21.7\%$	$+0.647$	$+0.663$	-2.5%
110	0.0653	0.0533	$+22.4\%$	$+0.646$	$+0.661$	-2.2%
150	0.0758	0.0602	$+25.9\%$	$+0.629$	$+0.633$	-0.7%
220	0.0870	0.0649	$+34.1\%$	$+0.610$	$+0.617$	-1.1%

C.3 EMBODIMENT MISMATCH BETWEEN REFERENCE AND TARGET

A useful offline metric should also fail when the reference data does not physically match the target task. We evaluate 3 bimanual WAMs on RoboTwin with two reference datasets that differ in arm configuration (Figure 12). With the bimanual ALOHA reference (Zhao et al., 2023), we find that S^2 recovers the success-rate order of all 3 models ($r=+0.999$). With the single-arm DROID reference, the correlation disappears ($r=+0.246$, $p=0.421$), since DTW over single-arm trajectories cannot capture bimanual motion. Different single-arm robots can still serve as references (Section 3.3), so what must match is the arm configuration, not the exact robot.

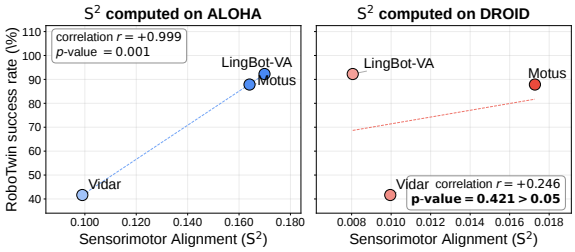


Figure 12: **S^2 requires matching arm configurations.** Evaluating 3 WAMs on bimanual RoboTwin, the bimanual ALOHA reference successfully recovers the success-rate ordering, while the single-arm DROID reference fails. S^2 tracks success only when reference and target have the same number of arms.

D Details of the Representation-Level Explanations

This section gives the full data behind the representation-level explanations in Section 4.

D.1 S^2 Across Feature Positions

We ask how VLA fine-tuning changes S^2 at each feature position. Table 4 reports the mean S^2 at each feature position for 3 groups of models on DROID ($N = 6{,}000$, $h = 7$), which are 3 standalone vision encoders, the 8 raw VLMs, and the 12 finetuned VLAs. Table 5 gives the per-model values. For the vision, text, and joint vision-text features of Section 4.1, we average-pool the selected output tokens and normalize the result. We find that fine-tuning raises the mean S^2 at every position. The gain is larger at the LLM output ($+29.9\%$ to $+45.0\%$) than at the vision-encoder output and LLM input ($+12.7\%$ to $+18.1\%$).

Table 4: **Mean S^2 per feature position** on DROID ($N = 6{,}000$, $h = 7$). Standalone vision encoders have only ViT-out features. The last column gives the relative VLM-to-VLA change.

Feature	Vision encoder (mean, $n = 3$)	Raw VLM (mean, $n = 8$)	Finetuned VLA (mean, $n = 12$)	Δ VLM→VLA
ViT-out	0.0246	0.0247	0.0278	+12.7%
LLM-in vision-only	–	0.0244	0.0275	+12.7%
LLM-in vision-text	–	0.0233	0.0275	+18.1%
LLM-out vision-only	–	0.0212	0.0275	+29.9%
LLM-out text-only	–	0.0140	0.0202	+45.0%
LLM-out vision-text	–	0.0204	0.0280	+37.4%

Table 5: **Per-model S^2 by feature position** on DROID ($N = 6{,}000$, $h = 7$). Means: Table 4.

Model	ViT-out	LLM-in vision-only	LLM-in text-only	LLM-in vision-text	LLM-out vision-only	LLM-out text-only	LLM-out vision-text
Pretrained vision encoders, no LLM attached ($n = 3$)							
CLIP-ViT-L/14	0.0240	–	–	–	–	–	–
SigLIP-SO400M	0.0247	–	–	–	–	–	–
SigLIP-2 Large	0.0250	–	–	–	–	–	–
Raw VLMs from VLM4VLA ($n = 8$)							
KosMos-2	0.0257	0.0238	0.0093	0.0239	0.0214	0.0151	0.0216
PaliGemma 1	0.0249	0.0251	0.0082	0.0193	0.0193	0.0160	0.0191
PaliGemma 2	0.0250	0.0251	0.0087	0.0222	0.0196	0.0142	0.0192
Qwen2.5-VL-3B	0.0241	0.0239	0.0079	0.0239	0.0185	0.0139	0.0188
Qwen2.5-VL-7B	0.0231	0.0234	0.0083	0.0234	0.0188	0.0127	0.0190
Qwen3-VL-2B	0.0242	0.0237	0.0089	0.0237	0.0236	0.0127	0.0217
Qwen3-VL-4B	0.0253	0.0254	0.0089	0.0254	0.0239	0.0135	0.0232
Qwen3-VL-8B	0.0250	0.0245	0.0088	0.0245	0.0240	0.0136	0.0207
Finetuned VLAs ($n = 12$)							
Qwen2.5-GR00T-Bridge	0.0344	0.0335	0.0078	0.0335	0.0354	0.0192	0.0363
Qwen2.5-GR00T-Bridge-RT-1	0.0355	0.0345	0.0077	0.0345	0.0350	0.0224	0.0353
Qwen2.5-OFT-Bridge-RT-1	0.0265	0.0265	0.0082	0.0265	0.0234	0.0185	0.0242
Qwen3-GR00T-Bridge-RT-1	0.0264	0.0266	0.0085	0.0266	0.0274	0.0158	0.0292
Qwen3-OFT-Bridge-RT-1	0.0263	0.0257	0.0086	0.0257	0.0248	0.0136	0.0254
CogACT Small	0.0301	0.0303	0.0097	0.0303	0.0313	0.0300	0.0319
CogACT Base	0.0301	0.0298	0.0096	0.0298	0.0319	0.0368	0.0324
CogACT Large	0.0298	0.0298	0.0097	0.0298	0.0317	0.0379	0.0328
GR00T-N1.5-Bridge	0.0252	0.0244	0.0090	0.0243	0.0238	0.0099	0.0240
GR00T-N1.6-Bridge	0.0251	0.0245	0.0090	0.0245	0.0258	0.0082	0.0255
Pi0-Bridge	0.0176	0.0178	0.0090	0.0180	0.0174	0.0161	0.0174
SpatialVLA-Bridge	0.0266	0.0263	0.0087	0.0265	0.0219	0.0145	0.0217

D.2 Frozen Versus Unfrozen Vision Encoder

Following VLM4VLA (Zhang et al., 2026a), both VLAs in Section 4.2 attach an FCDecoder action head to Qwen2.5-VL-3B (Bai et al., 2025b) and are fine-tuned on BridgeDataV2 with the same recipe, differing only in whether the vision encoder is frozen. We ask whether the S^2 drop from freezing the vision encoder depends on the trajectory horizon. Table 6 repeats the comparison at all 10 DTW

horizons, at the LLM-out vision-text feature read by the action head. We find that the frozen model has lower S^2 at every horizon, with relative reductions from 38.1% at $h=1$ to 49.4% at $h=220$.

Table 6: S^2 **with a frozen and an unfrozen vision encoder across horizons.** Qwen2.5-VL-3B VLAs trained with the same recipe, at the LLM-out vision-text feature on DROID ($N=6{,}000$).

Horizon h	**Unfrozen** S^2	**Frozen** S^2	Δ **relative**
1	0.0231	0.0143	-38.1%
3	0.0232	0.0141	-39.0%
7	0.0245	0.0144	-41.1%
15	0.0268	0.0158	-41.2%
25	0.0295	0.0166	-43.8%
40	0.0352	0.0191	-45.7%
75	0.0481	0.0258	-46.4%
110	0.0614	0.0326	-46.9%
150	0.0733	0.0390	-46.8%
220	0.0861	0.0435	-49.4%

D.3 WAM Features Along the Denoising Schedule

Table 7 reports S^2 for 3 WAMs at 7 feature positions and 5 points of each model's denoising schedule. Figure 10a plots the final-layer-norm row, and Figure 10b plots each row averaged over the schedule. Block 14 has the highest S^2 at 13 of the 15 model and schedule points.

Table 7: **Layer-wise** S^2 **along the denoising schedule for 3 WAMs.** Columns give each model's native denoising progress, from pure noise (0 %) to the clean prediction (100 %), on 50 RoboTwin tasks. Rows give 6 video-backbone positions and the text embedding. Block 14 (bold) has the highest S^2 at 13 of the 15 model and schedule points. The random baseline shuffles the same features.

Feature	0 % noise	25 %	50 %	75 %	100 % clean
Motus	random baseline $= 0.012 \pm 0.002$				
Block 0	0.044	0.203	0.307	0.321	0.324
Block 7	0.342	0.388	0.399	0.390	0.378
Block 14	**0.479**	**0.492**	**0.482**	**0.479**	**0.471**
Block 21	0.454	0.448	0.447	0.436	0.424
Block 29	0.420	0.424	0.419	0.421	0.405
Final layer norm	0.425	0.421	0.421	0.418	0.402
Text embedding	0.325	0.325	0.325	0.325	0.325
LingBot-VA	random baseline $= 0.012 \pm 0.001$				
Block 0	0.142	0.160	0.238	0.326	0.384
Block 7	0.404	0.399	0.431	0.449	0.455
Block 14	**0.498**	**0.504**	**0.514**	**0.514**	**0.513**
Block 21	0.422	0.435	0.428	0.445	0.449
Block 29	0.423	0.440	0.441	0.447	0.448
Final layer norm	0.425	0.429	0.431	0.439	0.451
Text embedding	0.355	0.355	0.355	0.355	0.355
Vidar	random baseline $= 0.013 \pm 0.004$				
Block 0	0.483	0.477	0.459	0.429	0.422
Block 7	0.467	0.484	0.485	0.479	0.469
Block 14	**0.504**	**0.555**	**0.531**	**0.518**	**0.473**
Block 21	0.508	0.510	0.500	0.488	0.475
Block 29	0.487	0.490	0.489	0.481	0.474
Final layer norm	0.487	0.491	0.491	0.481	0.476
Text embedding	0.467	0.467	0.467	0.467	0.467

Behavioral validation. To check that these differences relate to control, we vary video denoising steps in LingBot-VA (Li et al., 2026). We measure both S^2 and closed-loop success on the RoboTwin

`turn_switch` task over 100 episodes. Reducing the steps from the default 25 to 5 lowers success from 56% to 48% and S^2 from 0.288 to 0.241, so S^2 and control success change together.

E FULL S^2–SUCCESS RESULTS ACROSS HORIZONS

Tables 8 to 10 report S^2 for each of the 12 finetuned VLAs at all 10 DROID horizons ($N = 6{,}000$, $k = 10$), for the LLM output on vision-text, vision-only, and text-only input. Models are sorted by SimplerEnv WidowX success rate. The bottom rows give Spearman's ρ and Pearson's r with success, each with its one-tailed p. We find that S^2 from joint vision-text features and from vision features correlates significantly with success at every horizon ($r = +0.610$–$+0.717$ and $r = +0.554$–$+0.703$). Text-feature S^2 is not significant under Pearson's r at any horizon ($r = +0.227$–$+0.274$).

Table 8: S^2 from LLM-out joint vision-text features versus WidowX success rate across horizons, on DROID with $N = 6{,}000$, $k = 10$, and $n = 12$. Bold marks the horizon at which the across-model Pearson r peaks ($h = 7$).

Model	SR (%)	$h=1$	$h=3$	$h=7$	$h=15$	$h=25$	$h=40$	$h=75$	$h=110$	$h=150$	$h=220$
Qwen2.5-GR00T-Bridge	71.4	0.034	0.034	**0.036**	0.040	0.043	0.051	0.065	0.076	0.086	0.097
Qwen3-GR00T-RT-1-Bridge	65.3	0.028	0.027	**0.029**	0.032	0.036	0.044	0.057	0.071	0.083	0.096
Qwen2.5-GR00T-RT-1-Bridge	63.6	0.034	0.034	**0.035**	0.040	0.042	0.048	0.062	0.071	0.080	0.090
CogACT-Large	58.3	0.031	0.031	**0.033**	0.037	0.040	0.048	0.062	0.075	0.086	0.097
GR00T-N1.6-Bridge	57.1	0.023	0.024	**0.026**	0.029	0.031	0.038	0.051	0.064	0.076	0.091
CogACT-Base	51.3	0.031	0.031	**0.032**	0.037	0.040	0.049	0.062	0.074	0.086	0.098
CogACT-Small	51.0	0.031	0.031	**0.032**	0.036	0.040	0.047	0.060	0.071	0.083	0.092
Pi0-Bridge	47.9	0.017	0.017	**0.017**	0.020	0.022	0.025	0.033	0.044	0.053	0.061
Qwen2.5-OFT-RT-1-Bridge	44.2	0.024	0.023	**0.024**	0.028	0.030	0.036	0.049	0.061	0.070	0.082
Qwen3-OFT-RT-1-Bridge	42.7	0.024	0.024	**0.025**	0.029	0.031	0.038	0.051	0.064	0.075	0.086
SpatialVLA-Bridge	42.7	0.021	0.022	**0.022**	0.024	0.026	0.033	0.044	0.054	0.063	0.073
GR00T-N1.5-Bridge	36.5	0.023	0.023	**0.024**	0.027	0.030	0.036	0.048	0.058	0.069	0.080
Spearman ρ		0.736	0.764	**0.802**	0.788	0.767	0.767	0.774	0.746	0.774	0.725
p_{1t}		0.003	0.002	**0.001**	0.001	0.002	0.002	0.002	0.003	0.002	0.004
Pearson r		0.692	0.699	**0.717**	0.696	0.689	0.683	0.647	0.646	0.629	0.610
p_{1t}		0.006	0.006	**0.004**	0.006	0.007	0.007	0.012	0.012	0.014	0.018

Table 9: S^2 from LLM-out vision features versus WidowX success rate across horizons, on DROID with $N = 6{,}000$, $k = 10$, and $n = 12$. Bold marks the horizon of peak across-model Pearson r ($h = 15$).

Model	SR (%)	$h=1$	$h=3$	$h=7$	$h=15$	$h=25$	$h=40$	$h=75$	$h=110$	$h=150$	$h=220$
Qwen2.5-GR00T-Bridge	71.4	0.034	0.034	0.035	**0.039**	0.043	0.049	0.062	0.074	0.084	0.093
Qwen3-GR00T-RT-1-Bridge	65.3	0.026	0.026	0.027	**0.031**	0.034	0.041	0.054	0.067	0.080	0.092
Qwen2.5-GR00T-RT-1-Bridge	63.6	0.033	0.033	0.035	**0.038**	0.041	0.047	0.059	0.070	0.079	0.088
CogACT-Large	58.3	0.030	0.030	0.032	**0.035**	0.039	0.047	0.061	0.073	0.084	0.095
GR00T-N1.6-Bridge	57.1	0.024	0.024	0.026	**0.029**	0.032	0.039	0.051	0.064	0.077	0.091
CogACT-Base	51.3	0.031	0.030	0.032	**0.036**	0.040	0.048	0.060	0.073	0.085	0.098
CogACT-Small	51.0	0.030	0.030	0.031	**0.035**	0.039	0.046	0.059	0.070	0.082	0.092
Pi0-Bridge	47.9	0.017	0.017	0.017	**0.020**	0.022	0.025	0.033	0.044	0.053	0.061
Qwen2.5-OFT-RT-1-Bridge	44.2	0.022	0.022	0.023	**0.026**	0.028	0.035	0.046	0.058	0.068	0.079
Qwen3-OFT-RT-1-Bridge	42.7	0.024	0.024	0.025	**0.028**	0.030	0.037	0.050	0.063	0.074	0.087
SpatialVLA-Bridge	42.7	0.022	0.022	0.022	**0.025**	0.026	0.033	0.045	0.055	0.064	0.075
GR00T-N1.5-Bridge	36.5	0.023	0.023	0.024	**0.027**	0.029	0.035	0.048	0.059	0.069	0.079
Spearman ρ		0.767	0.757	0.767	**0.746**	0.746	0.739	0.725	0.711	0.683	0.704
p_{1t}		0.002	0.002	0.002	**0.003**	0.003	0.003	0.004	0.005	0.007	0.005
Pearson r		0.686	0.694	0.701	**0.703**	0.690	0.647	0.606	0.618	0.601	0.554
p_{1t}		0.007	0.006	0.006	**0.005**	0.007	0.011	0.018	0.016	0.019	0.031

Table 10: S^2 from LLM-out text features versus WidowX success rate across horizons, on DROID with $N = 6{,}000$, $k = 10$, and $n = 12$. Bold marks the horizon with the highest Spearman ρ.

Model	SR (%)	$h=1$	$h=3$	$h=7$	$h=15$	$h=25$	$h=40$	$h=75$	$h=110$	$h=150$	$h=220$
Qwen2.5-GR00T-Bridge	71.4	0.018	0.019	0.019	0.022	0.025	0.030	0.039	0.046	0.053	0.059
Qwen3-GR00T-RT-1-Bridge	65.3	0.015	0.015	0.016	0.018	0.020	0.024	0.031	0.040	0.048	0.057
Qwen2.5-GR00T-RT-1-Bridge	63.6	0.022	0.021	0.022	0.026	0.028	0.032	0.040	0.048	0.053	0.058
CogACT-Large	58.3	0.037	0.037	0.038	0.042	0.045	0.054	0.068	0.081	0.091	0.104
GR00T-N1.6-Bridge	57.1	0.008	0.008	0.008	0.010	0.010	0.014	0.020	0.024	0.029	0.033
CogACT-Base	51.3	0.035	0.035	0.037	0.041	0.044	0.053	0.068	0.080	0.090	0.104
CogACT-Small	51.0	0.030	0.029	0.030	0.034	0.038	0.044	0.057	0.064	0.070	0.080
Pi0-Bridge	47.9	0.014	0.015	0.016	0.019	0.021	0.024	0.033	0.043	0.052	0.061
Qwen2.5-OFT-RT-1-Bridge	44.2	0.019	0.018	0.018	0.022	0.024	0.028	0.036	0.044	0.051	0.060
Qwen3-OFT-RT-1-Bridge	42.7	0.013	0.013	0.014	0.015	0.016	0.020	0.026	0.034	0.040	0.045
SpatialVLA-Bridge	42.7	0.014	0.014	0.014	0.016	0.017	0.022	0.030	0.036	0.043	0.049
GR00T-N1.5-Bridge	36.5	0.010	0.010	0.010	0.011	0.011	0.014	0.019	0.024	0.029	0.033
Spearman ρ		0.438	0.452	0.452	0.452	0.452	0.452	0.501	0.452	**0.508**	0.291
p_{1t}		0.077	0.070	0.070	0.070	0.070	0.070	0.049	0.070	**0.046**	0.180
Pearson r		0.227	0.238	0.245	0.265	0.274	0.259	0.254	0.265	0.255	0.233
p_{1t}		0.239	0.229	0.222	0.203	0.194	0.208	0.213	0.202	0.212	0.233