WORLDORACLE: BENCHMARKING ACTION-CONDITIONED WORLD MODELS AGAINST REAL-WORLD EXECUTIONS

Anonymous authors
Paper under double-blind review

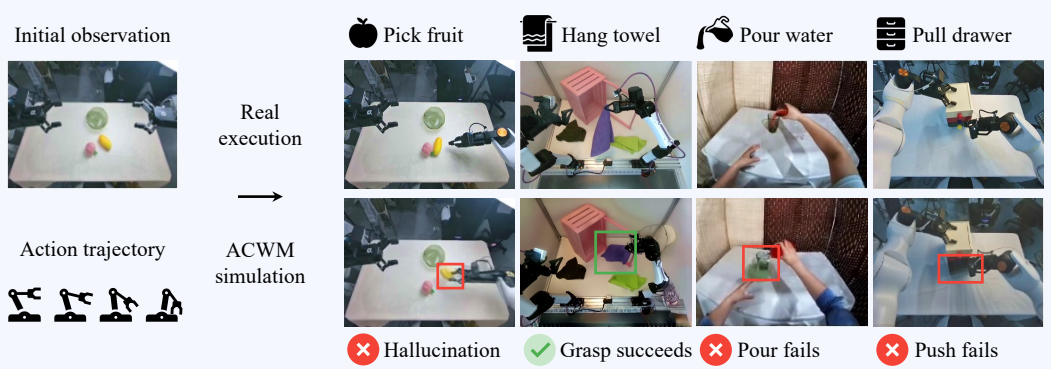

Figure 1: **Does the predicted future match reality?** Given the same initial observation and action trajectory, an action-conditioned world model (ACWM) can produce a visually plausible rollout that does not match what actually happens in the real world. In the "pick fruit" example, the robot fails to grasp the fruit, while the model hallucinates a successful grasp.

ABSTRACT

We introduce WorldOracle, a benchmark specifically designed to evaluate action-conditioned world models (ACWMs) against paired real-world executions. Given the same initial observation and low-level action sequence, WorldOracle asks whether a predicted rollout reproduces what happens in the real world, including physical consequences and task-state transitions. We evaluate models along four complementary dimensions: physical state correctness, interaction spatial alignment, key interaction area fidelity, and multi-view action coherence. To support this evaluation, we curate real-world training and evaluation suites that cover different interaction dynamics and embodiments. The evaluation suite contains long, multi-segment episodes with both successful and failed executions, enabling evaluation across varied action outcomes and behavior distributions. Experiments on seven representative ACWMs reveal distinct model strengths and failure modes. WorldOracle provides a systematic foundation for measuring and advancing action-conditioned world modeling for manipulation. We will release the datasets, evaluation code, and leaderboard.

1 INTRODUCTION

Recent advances in video generation (Agarwal et al., 2026; OpenAI et al., 2024; Google DeepMind, 2025; Wan et al., 2025; Yang et al., 2025b) have opened a new path toward learned world simulation for robotic manipulation. Trained on large-scale interaction data, these models can capture a wide range of scenes, objects, and contact-rich behaviors that are difficult to model with hand-crafted simulators. Recent work has therefore adapted video generators into action-conditioned world models (ACWMs) (Qiu et al., 2026; Jiang et al., 2025; Gao et al., 2026; Guo et al., 2025; Zhu et al., 2024; Tseng et al., 2026). Given an initial observation and a low-level robot action sequence, an ACWM

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

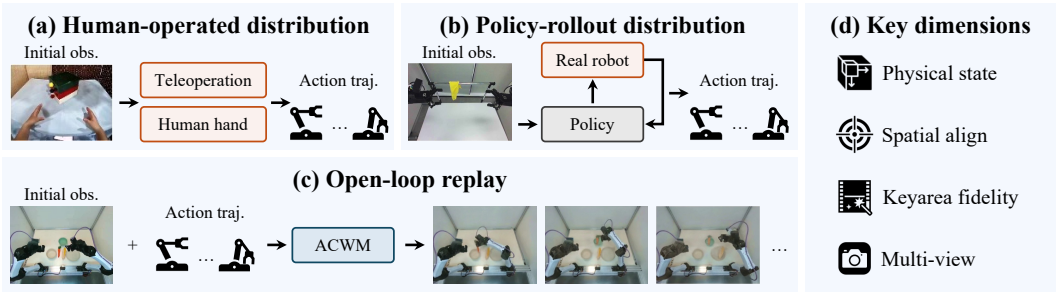

Figure 2: **Overview of WorldOracle.** **(a)** The human-operated trajectories are collected through robot teleoperation or direct human-hand execution. **(b)** The policy-rollout trajectories are collected by running a policy in closed loop on a real robot. **(c)** For both distributions, the recorded action trajectory is replayed open loop by the ACWM, yielding a generated rollout. **(d)** The predicted rollout is compared with the paired real execution along four dimensions: physical state correctness, interaction spatial alignment, key interaction area fidelity, and multi-view action coherence.

predicts how the scene will evolve as the robot executes those actions. Such models can potentially support data generation, scalable policy evaluation, and closed-loop policy learning. But for these applications, producing a realistic future is not enough. The predicted future should reproduce what actually happens when the same actions are executed in the real world. As illustrated in Fig. 1, a rollout can look visually convincing while still getting the outcome wrong. In the "pick fruit" example, the real gripper misses the fruit, but the model hallucinates a successful grasp.

We refer to the agreement between a predicted rollout and its paired real-world execution as *execution coherence*. A reliable ACWM should maintain this coherence under three basic stress tests. First, *causal obedience* asks whether the prediction follows the given action sequence, including incorrect actions and failed attempts. Second, *distribution transport* asks whether this agreement holds across different embodiments and action distributions produced by human operators or autonomous policies. Third, *temporal endurance* asks whether this coherence can be maintained over long rollouts, rather than only over short prediction horizons.

Existing benchmarks do not jointly test these properties against physical execution. General world-model benchmarks (Duan et al., 2025; Li et al., 2026) assess visual quality, semantic alignment, controllability, and physical plausibility, but do not target embodied action-conditioned prediction. Embodied world-model benchmarks (Yue et al., 2025; Deng et al., 2026; Chen et al., 2026; Zhou et al., 2026; Fan et al., 2026; Jiang et al., 2026) evaluate manipulation videos conditioned on text or task-level instructions, but do not directly test whether a predicted future matches the physical consequences of a specific low-level action trajectory. More recent action- and executability-oriented benchmarks (Shang et al., 2026a;b) evaluate embodied world models as data engines, policy evaluators, planners, or interactive learning environments. These evaluations, however, rely mostly on simulation. As a result, they do not directly measure whether an ACWM reproduces what happens when the same actions are executed on a real robot, especially over long trajectories that include both successful and failed or counterfactual behaviors.

To close this gap, we introduce WorldOracle, a benchmark for evaluating ACWMs against paired real-world executions. Its central question is simple: given the same initial observation and low-level action trajectory, does the predicted future match what actually happens in the real world?

As illustrated in Fig. 2, WorldOracle is designed around the three stress tests introduced above: causal obedience is tested by replaying the same action trajectory in the ACWM and comparing it with the paired real execution; distribution transport is tested across embodiments and action sources; and temporal endurance is tested over long, multi-segment rollouts. **Causal obedience.** We evaluate whether an ACWM follows the given action sequence even when the execution is imperfect or unsuccessful. Each episode is divided into temporally ordered *interaction segments*, where each segment corresponds to one manipulation attempt and ends at an *outcome keyframe* that records its physical result, such as a successful lift, an empty grasp, or a dropped object (Fig. 1). This lets us compare the predicted and real outcomes of individual attempts, including failures. **Distribution transport.** We evaluate whether execution coherence transfers across embodiments and action distributions. Our benchmark includes AgiBot G2, YAM, and human-hand embodiments, covering different robot morphologies and interaction styles. We also include a policy-rollout split

on AgiBot G2, where actions are generated by a closed-loop policy. These rollouts introduce a different action distribution, including hesitation, jitter, and corrections. **Temporal endurance.** We evaluate long-duration rollouts, testing whether execution coherence is maintained as prediction errors accumulate over time.

For each episode, the ACWM receives the recorded initial observation and low-level action trajectory and replays the actions open loop to generate a predicted rollout. We then compare this rollout with its paired real-world reference along four dimensions. *Physical state correctness* measures whether the predicted outcome matches the real physical state. *Interaction spatial alignment* measures whether the robot and manipulated objects remain aligned with their real-world trajectories. *Key interaction area fidelity* evaluates visual fidelity in the interaction regions, including manipulated objects and robot arms or human hands. For models that support multi-view outputs, *multi-view action coherence* tests whether the predicted robot execution is aligned across synchronized camera views.

To support this evaluation, we curate real-world training and evaluation suites across AgiBot G2, YAM, and human-hand embodiments. The training suite includes a newly collected AgiBot G2 dataset, supplemented with the public ABC-130K dataset (Allshire et al., 2026). The evaluation suite contains human-operated rollouts for all three embodiments and a policy-rollout split on AgiBot G2. As shown in Fig. 1, the benchmark covers four tasks with different interaction dynamics: fruit picking for rigid-body manipulation, towel hanging for deformable objects, water pouring for fluid dynamics, and drawer pulling for articulated objects. Each episode contains multiple consecutive interaction segments, enabling evaluation of repeated manipulation attempts over long rollouts. The datasets also include both successful and failed interaction segments. This lets WorldOracle test whether ACWMs follow the given actions across different outcomes, embodiments, and rollout lengths.

We also build an annotation pipeline for the evaluation suite. As shown in Fig. 3, human annotators mark interaction-segment boundaries and their outcome keyframes. A VLM then generates physical-state criteria for each keyframe, such as "the green cup is grasped by the gripper". These criteria are used to evaluate physical state correctness. We use SAM3 (Carion et al., 2026) to obtain frame-wise masks of manipulated objects, robot arms, and human hands. These masks focus the evaluation on action-relevant regions instead of the background, and are used for spatial matching and local visual comparison in interaction spatial alignment and key interaction area fidelity. We apply the same annotation procedure to synchronized camera views for multi-view action coherence.

We evaluate seven representative models across metrics, tasks, embodiments, and rollout lengths. The experiments reveal several failure modes. Visually plausible rollouts can misrepresent object dynamics and interaction outcomes, even when robot-arm motion is accurately reproduced. Strong short-horizon performance also does not consistently translate into reliable long-horizon prediction. We also observe performance patterns associated with different modeling choices, which we discuss as hypotheses rather than causal conclusions. In summary, our contributions are threefold:

- We introduce WorldOracle, a benchmark for evaluating ACWM execution coherence against paired real-world rollouts under the same initial observations and low-level action trajectories.
- We curate real-world training and evaluation suites spanning diverse interaction dynamics, multiple embodiments, human-operated and policy-rollout trajectories, and long episodes with both successful and failed executions.
- We provide an evaluation toolkit and leaderboard covering four complementary metrics, and use them to identify recurring failure modes and performance patterns across current ACWMs.

2 Related Work

2.1 Embodied World Models

Recent video generators have motivated embodied world models (EWMs) that serve as learned visual simulators for robotic interaction (Agarwal et al., 2026; Wan et al., 2025; Google DeepMind, 2025). Early EWMs predict future observations from an initial image and a high-level instruction, whereas action-conditioned world models (ACWMs) introduce low-level actions to control the generated rollouts (Zhu et al., 2024; Guo et al., 2025; Agarwal et al., 2026; Gao et al., 2026; Qiu et al., 2026).

ACWMs differ primarily in their action injection mechanisms. IRASim (Zhu et al., 2024) aligns frame-level action embeddings with generated observations. Ctrl-World (Guo et al., 2025) supports

Benchmark	# Samples	Action.	Real.	Fail.	Long.	Multi-Embod.	Multi-view.	Policy Dist.
EWMBench (Yue et al., 2025)	100	✗	✓	✗	✗	✗(AgiBot G1)	✗	✗
PAI-Bench (Zhou et al., 2026)	2808	✗	✓	✗	✗	✓(AgiBot G1, WidowX, Human hand)	✗	✗
WoW-World-Eval (Fan et al., 2026)	609	✗	✓	✗	✗	✓(Franka, UR5e, AgileX, humanoid)	✗	✗
RBench (Deng et al., 2026)	650	✗	✓	✗	✗	✓(single-, dual-arm,humanoid, quadruped)	✗	✗
WorldArena (Shang et al., 2026a)	500	✓	✗	✗	✗	✗(RoboTwin)	✗	✓
EZSBench (Chen et al., 2026)	196	✗	✓	✗	✗	✓(real/synthetic robot morphologies)	✗	✗
RoboWM-Bench (Jiang et al., 2026)	160	✗	✓	✓	✗	✓(Franka, Human hand)	✗	✗
WorldArena 2.0 (Shang et al., 2026b)	500+	✓	✗	✗	✗	✓(RoboTwin, LIBERO/Panda, AgileX)	✗	✓
WorldOracle (Ours)	335	✓	✓	✓	✓	✓(AgiBot G2, YAM, Human hand)	✓	✓

Table 1: **Comparison of Embodied World-Model Benchmarks.** *# Samples*: number of samples for evaluation. *Action.*: evaluation of models conditioned on action sequences. *Real.*: real-world data. *Fail.*: data with failed executions. *Long.*: long-duration execution traces with multiple interaction segments. *Multi-Embod.*: evaluation across multiple embodiments. *Multi-view.*: evaluation across multiple camera views. *Policy Dist.*: action trajectories generated by autonomous policy rollouts.

joint multi-view rollouts with memory retrieval, while ORV (Yang et al., 2025a) and Kinema4D (Xu et al., 2026) introduce occupancy- and kinematics-based geometric guidance. DreamDojo (Gao et al., 2026) transfers latent actions learned from human videos to continuous robot actions, and Cosmos 3 (Agarwal et al., 2026) unifies video and action across heterogeneous spaces. GE-Sim 2.0 (Qiu et al., 2026) further advances closed-loop simulation by jointly predicting proprioceptive states.

WorldOracle provides an evaluation suite that allows each ACWM to use its native conditioning interface while being evaluated against paired real-world execution references.

2.2 Embodied World Model Benchmarks

A growing number of benchmarks evaluate generated videos in embodied settings (Yue et al., 2025; Zhou et al., 2026; Fan et al., 2026; Deng et al., 2026; Shang et al., 2026a;b; Chen et al., 2026; Jiang et al., 2026). EWMBench (Yue et al., 2025), RBench (Deng et al., 2026), and EZS-Bench (Chen et al., 2026) assess properties such as visual quality, motion, physical consistency, task completion, and cross-embodiment generalization. PAI Bench (Zhou et al., 2026) and WoW-World-Eval (Fan et al., 2026) further consider physical understanding, planning, plausibility, and executability. However, these benchmarks primarily evaluate videos conditioned on initial image and a task-level instruction.

More recent benchmarks consider embodied execution. WorldArena (Shang et al., 2026a) evaluates world models as data engines, policy evaluators, and action planners, but conducts its functional evaluation in the simulator. WorldArena 2.0 (Shang et al., 2026b) extends this setting to visuotactile prediction, online policy learning, and cross-platform transfer, while remaining predominantly simulation-based. RoboWM-Bench (Jiang et al., 2026) evaluates physical executability by extracting actions from generated videos and replaying them in real-to-sim environments. Despite these advances, none directly assess whether action-conditioned predictions are coherent with corresponding real-world executions. We show a detailed comparison with existing benchmarks in Table 1.

3 The WorldOracle Benchmark

Problem formulation. We represent episode i as $\mathcal{E}_i = (\mathbf{O}_i, \mathbf{A}_i)$. The observation sequence $\mathbf{O}_i = \left(\{O^v_{i,t}\}_{v\in\mathcal{V}}\right)_{t=0}^{T_i}$ contains synchronized observations from the camera views $\mathcal{V} = \{\texttt{top_head}, \texttt{hand_left}, \texttt{hand_right}\}$. We use `top_head` as the primary view, while `hand_left` and `hand_right` are the left and right hand-mounted camera views. The corresponding low-level action sequence is denoted by $\mathbf{A}_i = (A_{i,t})_{t=1}^{T_i}$. Each episode is divided into N_i temporally ordered interaction segments $\{\mathcal{S}_{i,n}\}_{n=1}^{N_i}$. Each segment corresponds to one manipulation attempt and spans $\mathcal{S}_{i,n} = [t^{\text{start}}_{i,n}, t^{\text{end}}_{i,n}]$. We define its outcome keyframe as the final observation of the segment: $K^v_{i,n} := O^v_{i,t^{\text{end}}_{i,n}}$. The outcome keyframe captures the observable physical result of the interaction. For each evaluation episode i, WorldOracle provides the initial observation $\left\{O^v_{i,0}\right\}_{v\in\mathcal{V}}$ and the recorded real-world action sequence $\mathbf{A}_i$. An ACWM $\mathcal{W}_\theta$ generates the predicted rollout:

$$\hat{\mathbf{O}}_i = \mathcal{W}_\theta\left(\left\{O^v_{i,0}\right\}_{v\in\mathcal{V}}, \mathbf{A}_i\right), \tag{1}$$

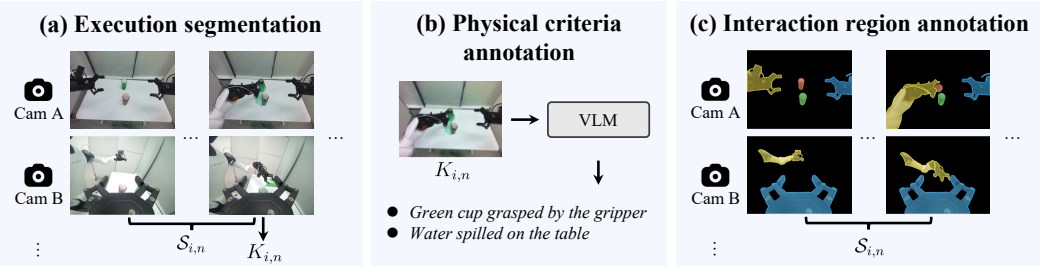

Figure 3: **WorldOracle annotation pipeline. (a)** Execution segmentation: each synchronized multi-view episode is divided into interaction segments $\mathcal{S}_{i,n}$, and the final frame of each segment is the outcome keyframe $K_{i,n}$ capturing the observable physical outcome. **(b)** Physical criteria annotation: a VLM generates physical-state criteria for each outcome keyframe. **(c)** Interaction region annotation: manipulated objects, robot arms, and human hands are annotated frame-wise across camera views.

which is compared with the paired real-world rollout $\mathbf{O}_i$. Because both rollouts start from the same observation and follow the same action sequence, this comparison directly tests whether the ACWM reproduces the physical consequences of those actions correctly.

Design overview. WorldOracle is designed to support ACWMs with various low-level action conditioning and injection mechanisms. We design benchmark tasks based on two principles. First, they should cover different types of interaction dynamics, including rigid-body, deformable-object, fluid, and articulated-object interactions. Second, both successful and failed executions should lead to physical outcomes that are directly observable and visually unambiguous. This makes it possible to evaluate task outcomes reliably with a VLM. These principles lead to four representative tasks shown in Fig. 1: *pick fruit*, *hang towel*, *pour water*, and *pull drawer*.

We next describe the training and evaluation suites, the annotation pipeline, and the four evaluation metrics used to quantify execution coherence.

3.1 Data Curation

Training suite. The training suite is used to adapt models to the target embodiments before evaluation on the held-out test set. For AgiBot G2, we collect a new real-world dataset with 1,779 episodes across the aforementioned four benchmark tasks. Each episode lasts about one minute and contains multiple interaction segments. Around 20% of the segments in each episode are failed or suboptimal executions. This composition enables ACWMs to learn the actual physical consequences of erroneous actions instead of associating every action sequence with successful task completion. For YAM, we use ABC-130K (Allshire et al., 2026), a large public dataset of bimanual manipulation collected on YAM stations. We do not provide adaptation data for the human-hand embodiment; models with egocentric world-modeling capabilities are evaluated zero-shot instead. More details are provided in App. E.1 and Table S1.

Human-operated distribution. The human-operated evaluation split contains trajectories collected through robot teleoperation or direct human-hand execution (Fig. 2(a)). We collect held-out real-world episodes for AgiBot G2, YAM, and human-hand embodiments, with 100, 75, and 100 episodes, respectively. Each episode lasts about two minutes and contains multiple consecutive interaction segments, including both successful and failed manipulation attempts. Compared with the training suite, the evaluation suite is longer and contains a higher proportion of failures, with approximately 40% of the segments corresponding to failed executions for each episode. This increased failure frequency tests whether an ACWM can reproduce the consequences of non-ideal actions instead of defaulting to successful task completion.

We design each interaction segment to end in a physical outcome that is clearly visible and easy for a VLM to judge. For example, the VLM evaluator can directly determine whether a fruit is held by the gripper or placed in a target container, whether liquid is transferred or spilled, whether a drawer is open or closed, and whether a towel is hanging over the rack or remains on the table. This enables reliable VLM-assisted assessment during the evaluation. Details are provided in App. E.2 and Table S2.

Policy-rollout distribution. We additionally construct a policy-rollout evaluation split on the physical AgiBot G2 robot (Fig. 2(b)). A $\pi_{0.5}$ policy (Physical Intelligence et al., 2025) trained on our curated training data interacts with the real robot in closed loop. During evaluation, each recorded action trajectory is replayed open loop from its paired initial observation (Fig. 2(c)). This split contains 60 episodes. More details are provided in App. E.3 and App. H.

Annotation pipeline. We build a systematic annotation pipeline for evaluation suites. As illustrated in Fig. 3, each evaluation episode is temporally divided into interaction segments. Each segment terminates at an outcome keyframe that captures an observable successful or failed physical outcome. For physical criteria annotation, a VLM-assisted pipeline generates physical-state criteria for each outcome keyframe. For interaction region annotation, we use the segmentation method to obtain frame-wise masks of manipulated objects and robot arms or human hands. (See App. F for details).

3.2 THE WORLDORACLE METRICS

Given a real observation sequence $\mathbf{O}_i = (\mathbf{O}_{i,t})_{t=0}^{T_i}$ and a prediction $\hat{\mathbf{O}}_i = (\hat{\mathbf{O}}_{i,t})_{t=0}^{T_i}$ conditioned on the same initial observation and action trajectory, WorldOracle evaluates execution coherence along four dimensions. Physical state correctness, interaction spatial alignment, and key interaction area fidelity are evaluated on the primary `top_head` view. Multi-view action coherence evaluates all three synchronized views in $\mathcal{V}$. Full metric definitions and details are provided in App. G.

Physical state correctness. A rollout can look realistic while still predicting the wrong physical outcome. For example, the model may hallucinate a successful grasp of an object even when the given action trajectory causes the real gripper to close without picking up the object (Fig. 1). Therefore, for each interaction segment, we evaluate the predicted outcome keyframe against physical-state criteria annotated from the paired real-world keyframe (Fig. 3(b)). A VLM assigns a binary judgement to each criterion. The resulting binary criterion scores are averaged across criteria for each segment, and the episode score is obtained by averaging over its execution segments. This metric evaluates whether the predicted execution reaches the same physical outcome as the real execution.

Interaction spatial alignment. Matching the final physical outcome does not necessarily mean that an ACWM follows the correct motion throughout the execution. We therefore introduce interaction spatial alignment, which measures whether robot arms or human hands and manipulated entities remain spatially aligned with their real-world trajectories. At each sampled frame, DINOv3 features (Siméoni et al., 2025) within the annotated arm or hand and object masks (Fig. 3(c)) are matched to features in the predicted observations. The score combines the spatial displacement of these matches with their correspondence confidence, assigning a high score when the action-relevant entities are reliably identified near their real-world positions over time.

Key interaction area fidelity. Whole-frame perceptual metrics can be dominated by background pixels and overlook errors in regions where interactions actually happen. We therefore use masks from the real execution to identify the expected regions of manipulated objects and robot arms or human hands (as shown in Fig. 3(c)), and evaluate fidelity only within these regions. At each sampled frame, the corresponding masked regions are cropped from the predicted and real observations. We then evaluate their pixel-level, perceptual, and semantic appearance fidelity using PSNR (Huynh-Thu & Ghanbari, 2008), LPIPS (Zhang et al., 2018), and V-JEPA similarity (Assran et al., 2025), respectively. After normalization, the three components are combined using a geometric mean, so a severe error in any component lowers the final score. A high score requires the relevant entity to appear in the expected region and to match the paired real execution in appearance. This also penalizes spatial drift outside the expected region. The metric captures local differences such as object deformation, gripper artifacts, and incorrect contact geometry.

Multi-view action coherence. For multi-view models, multi-view action coherence measures robot execution alignment with the paired real execution across all synchronized camera views. It is reference-grounded rather than a direct measure of intrinsic geometric consistency between generated views, as structure-from-motion estimates for robot arms are unreliable. We therefore compute robot-arm alignment independently in each view using a similar correspondence strategy as interaction spatial alignment, and then aggregate the per-view scores. A frame receives a high score only when the predicted robot motion is well aligned in every view. This penalizes cases where the robot appears correctly aligned in one view but is missing or spatially misplaced in another.

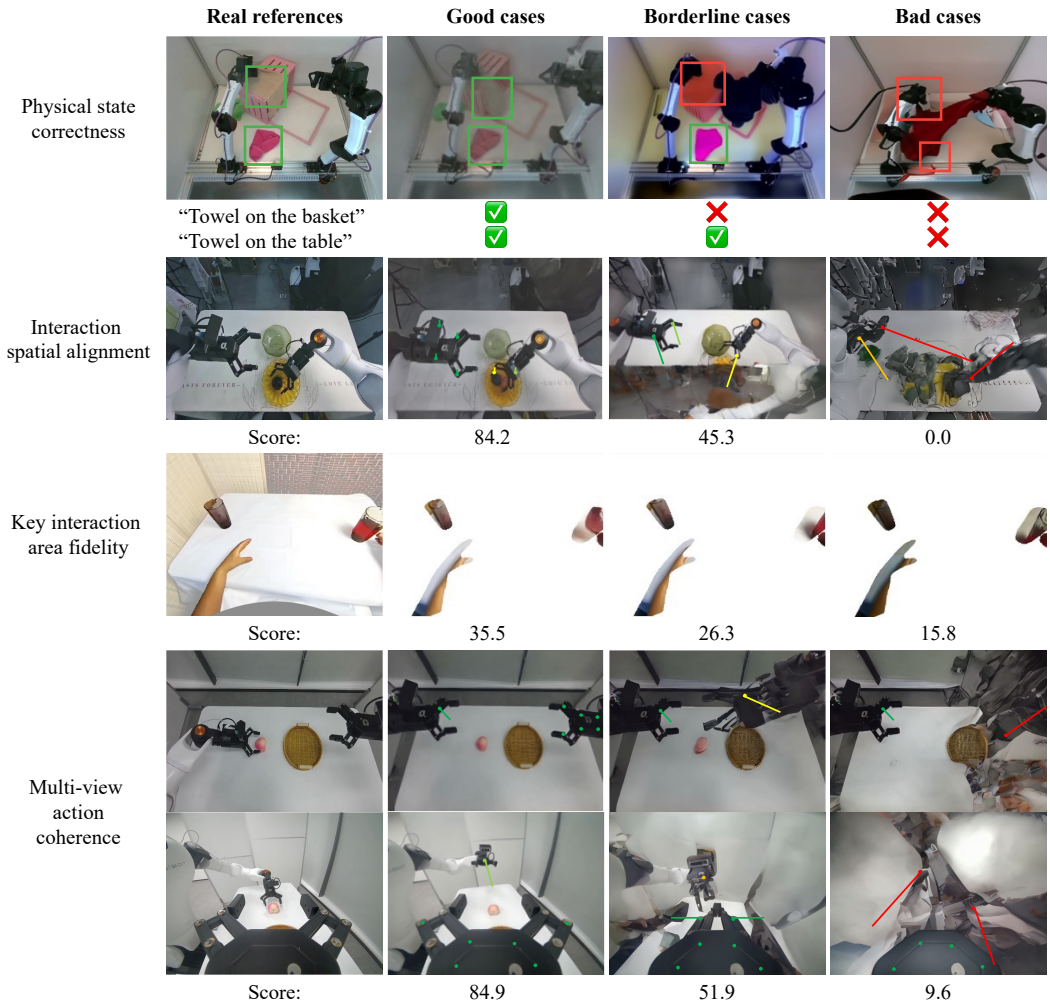

Figure 4: **Qualitative examples.** Each row compares a real-world reference with high-, intermediate-, and low-scoring predictions. *Physical state correctness* checks whether the prediction satisfies the criteria. The bad case predicts states of both towels wrong. *Interaction spatial alignment.* Points indicate matched visual features and lines show their displacement to reference. Point color denotes matching confidence from green (high) to red (low), while shorter lines indicate better spatial alignment. *Key interaction area fidelity.* Within the interaction regions, the predicted entities of the bad example exhibit degraded appearance and structural distortion. *Multi-view action coherence.* In the bad example, the predicted robot arm is poorly aligned in the synchronized camera views.

Score normalization and aggregation. All four dimensions are reported on a common $[0, 100]$ scale, using the metric-specific scoring and normalization procedures described in App. G. Let S_{phys}, S_{isa}, S_{key}, S_{mv} denote the scores for physical state correctness, interaction spatial alignment, key interaction area fidelity, and multi-view action coherence, respectively. The single-view track aggregates the three dimensions evaluated on the primary `top_head` view:

$$\text{WorldOracle-SV} = \frac{S_{\text{phys}} + S_{\text{isa}} + S_{\text{key}}}{3}. \tag{2}$$

For multi-view predictions, the multi-view track additionally includes multi-view action coherence:

$$\text{WorldOracle-MV} = \frac{S_{\text{phys}} + S_{\text{isa}} + S_{\text{key}} + S_{\text{mv}}}{4}. \tag{3}$$

All dimensions receive equal weight within their respective tracks. We further report scores under two temporal regimes. The short-duration regime includes sampled frames and outcome keyframes within the first 30 seconds. We choose this boundary based on the empirical duration distribution of

Table 2: **Results on AgiBot G2.** The 30s and Long rows evaluate the first 30 seconds and the complete episode. All metrics are reported on a $[0, 100]$ scale; higher is better.

Model	Regime	Physical Correctness	Spatial Alignment	Key Area Fidelity	Multi-View Coherence	WorldOracle	
						-SV	-MV
IRASim (Zhu et al., 2024)	30s	$\underline{55.64} \pm 3.16$	53.96 ± 2.10	38.00 ± 1.26	–	49.20 ± 1.31	–
	Long	$\underline{42.06} \pm 2.03$	$\underline{37.60} \pm 1.72$	$\underline{25.47} \pm 1.00$	–	$\underline{35.04} \pm 0.95$	–
Ctrl-World (Guo et al., 2025)	30s	51.15 ± 3.47	57.08 ± 2.40	40.53 ± 1.51	$\underline{65.57} \pm 2.05$	49.59 ± 1.47	$\underline{53.58} \pm 1.19$
	Long	38.31 ± 2.96	29.44 ± 2.34	23.45 ± 1.23	$\underline{39.31} \pm 1.88$	30.40 ± 1.32	$\underline{32.62} \pm 1.10$
DreamDojo (Gao et al., 2026)	30s	53.32 ± 3.63	53.96 ± 1.71	$\underline{44.39} \pm 1.31$	–	50.56 ± 1.41	–
	Long	37.97 ± 2.30	18.10 ± 1.01	25.18 ± 0.92	–	27.08 ± 0.87	–
Cosmos 3 (Agarwal et al., 2026)	30s	53.12 ± 3.07	$\underline{67.41} \pm 1.52$	41.44 ± 1.08	–	$\underline{53.99} \pm 1.17$	–
	Long	33.43 ± 1.71	25.26 ± 1.05	21.10 ± 0.84	–	26.60 ± 0.72	–
GE-Sim 2.0 (Qiu et al., 2026)	30s	$\mathbf{72.54} \pm 3.52$	$\mathbf{81.40} \pm 1.34$	$\mathbf{67.62} \pm 1.04$	$\mathbf{97.53} \pm 0.25$	$\mathbf{73.86} \pm 1.27$	$\mathbf{79.77} \pm 0.95$
	Long	$\mathbf{60.26} \pm 2.79$	$\mathbf{73.49} \pm 1.77$	$\mathbf{56.24} \pm 0.90$	$\mathbf{96.96} \pm 0.39$	$\mathbf{63.33} \pm 1.11$	$\mathbf{71.74} \pm 0.85$

Table 3: **Results on the AgiBot G2 policy-rollout distribution.** Each rollout is evaluated over its complete duration. All scores are reported on a $[0, 100]$ scale, and higher is better.

Model	Physical Correctness	Spatial Alignment	Key Area Fidelity	Multi-View Coherence	WorldOracle	
					-SV	-MV
IRASim (Zhu et al., 2024)	$\mathbf{62.18} \pm 7.04$	51.62 ± 3.74	32.30 ± 2.19	–	$\underline{48.70} \pm 2.67$	–
Ctrl-World (Guo et al., 2025)	27.22 ± 6.52	42.95 ± 3.81	33.16 ± 2.52	$\underline{62.91} \pm 3.69$	34.44 ± 2.69	$\underline{41.56} \pm 2.20$
DreamDojo (Gao et al., 2026)	44.44 ± 7.86	52.88 ± 3.53	$\underline{36.57} \pm 2.52$	–	44.63 ± 2.97	–
Cosmos 3 (Agarwal et al., 2026)	34.29 ± 6.78	$\underline{59.81} \pm 3.02$	34.31 ± 2.81	–	42.80 ± 2.60	–
GE-Sim 2.0 (Qiu et al., 2026)	$\underline{57.03} \pm 6.89$	$\mathbf{78.70} \pm 1.84$	$\mathbf{58.33} \pm 1.39$	$\mathbf{96.64} \pm 0.35$	$\mathbf{64.69} \pm 2.37$	$\mathbf{72.67} \pm 1.79$

existing manipulation datasets; details are provided in App. C. The long-episode regime evaluates all available frames and outcome keyframes. Models with synchronized multi-view outputs are evaluated on both tracks, while single-view models are evaluated only on WorldOracle-SV.

4 RESULTS

Validation. We validate WorldOracle through human studies and controlled analyses. The metric-induced preferences agree closely with human judgements in 88.1–93.2% of pairwise comparisons, with Spearman correlations ranging from 0.72 to 0.82. We show more details in App. B.1. We also verify the annotated physical criteria on their real outcome keyframes. The VLM judge marks all criteria as satisfied, yielding a criterion–reference agreement of 100%. Independent human annotators further check the criteria against the reference images, to avoid relying only on VLM self-consistency. Controlled examples show that physical state correctness and interaction spatial alignment capture complementary failure modes, while key interaction area fidelity decreases monotonically under increasing corruption inside the interaction regions but remains unchanged under background-only corruption. Further validation details are provided in App. B.

Models and implementation details. We evaluate seven ACWMs: IRASim, Ctrl-World, DreamDojo, Cosmos 3 Nano, GE-Sim 2.0, JointControlVideo, and OSCAR. For each model, we start from the official pretrained checkpoint and follow its original training setup, action representation, and conditioning mechanism. The robot-oriented models are adapted to AgiBot or YAM using our AgiBot training suite or the public ABC-130K dataset, while models with egocentric world-modeling capabilities are evaluated zero-shot on the human-hand embodiment. To collect the policy-rollout evaluation split, we initialize a $\pi_{0.5}$ policy from its official DROID checkpoint, fine-tune it on the AgiBot training suite, and deploy it on the physical robot. Complete training and implementation details are provided in App. H.

4.1 OBSERVATIONS AND CHALLENGES

The AgiBot G2, YAM, and human-hand evaluations should not be interpreted as a controlled comparison of embodiment difficulty. The models are adapted differently across settings: AgiBot G2 uses a benchmark-specific training suite, YAM relies on the public ABC-130K dataset and therefore has a larger train–test gap, and the human-hand models are evaluated zero-shot. We therefore compare models primarily within each setting. Results are shown in Tables 2–5. Each score is reported as the mean $\pm$ 95% bootstrap confidence interval, estimated from 5,000 episode resamples. More results and analyses are provided in App. A.

Table 4: **Results on YAM.** The 30s and Long rows report short-duration and full-episode evaluation, respectively. All metrics are reported on a $[0, 100]$ scale, and higher is better.

Model	Regime	Physical Correctness	Spatial Alignment	Key Area Fidelity	Multi-View Coherence	WorldOracle	
						-SV	-MV
IRASim (Zhu et al., 2024)	30s	65.23 ± 4.69	51.52 ± 1.96	39.67 ± 1.13	–	52.14 ± 1.74	–
	Long	$\underline{53.39} \pm 2.75$	$\underline{38.71} \pm 1.60$	$\underline{30.35} \pm 0.93$	–	$\underline{40.82} \pm 1.11$	–
Ctrl-World (Guo et al., 2025)	30s	56.39 ± 4.88	53.43 ± 2.70	35.93 ± 1.58	$\underline{41.87} \pm 2.15$	48.59 ± 1.91	$\underline{46.91} \pm 1.53$
	Long	39.03 ± 3.02	20.71 ± 2.04	18.97 ± 1.35	$\underline{15.52} \pm 1.34$	26.24 ± 1.28	$\underline{23.56} \pm 1.02$
DreamDojo (Gao et al., 2026)	30s	59.14 ± 4.31	56.35 ± 2.78	$\underline{45.84} \pm 1.65$	–	53.78 ± 1.79	–
	Long	48.15 ± 3.09	19.70 ± 2.11	28.11 ± 1.13	–	31.99 ± 1.27	–
Cosmos 3 (Agarwal et al., 2026)	30s	$\mathbf{67.84} \pm 4.13$	$\underline{68.93} \pm 2.14$	40.71 ± 1.18	–	$\underline{59.16} \pm 1.60$	–
	Long	$\mathbf{62.85} \pm 2.52$	34.61 ± 1.83	16.04 ± 0.74	–	37.83 ± 1.08	–
GE-Sim 2.0 (Qiu et al., 2026)	30s	$\underline{67.78} \pm 5.35$	$\mathbf{74.97} \pm 1.97$	$\mathbf{54.64} \pm 1.11$	$\mathbf{64.10} \pm 2.17$	$\mathbf{65.80} \pm 1.92$	$\mathbf{65.37} \pm 1.54$
	Long	49.37 ± 3.17	$\mathbf{59.78} \pm 2.37$	$\mathbf{44.41} \pm 1.22$	$\mathbf{48.54} \pm 1.56$	$\mathbf{51.19} \pm 1.39$	$\mathbf{50.53} \pm 1.11$

Table 5: **Results on Human Hand.** The 30s and Long rows report short-duration and full-episode evaluation, respectively. All metrics are reported on a $[0, 100]$ scale, and higher is better.

Model	Regime	Physical Correctness	Spatial Alignment	Key Area Fidelity	WorldOracle-SV
JointControlVideo (Zhang et al., 2026)	30s	$\mathbf{55.47} \pm 4.24$	$\mathbf{57.24} \pm 2.46$	20.50 ± 1.40	$\mathbf{44.40} \pm 1.69$
	Long	$\mathbf{56.33} \pm 3.02$	$\mathbf{23.79} \pm 1.66$	8.10 ± 0.67	$\mathbf{29.41} \pm 1.18$
Cosmos 3 (Agarwal et al., 2026)	30s	47.46 ± 3.88	33.70 ± 2.33	$\underline{22.21} \pm 1.06$	34.46 ± 1.58
	Long	30.23 ± 2.22	9.43 ± 0.87	$\underline{11.44} \pm 0.72$	17.03 ± 0.82
OSCAR (Wu & Gao, 2026)	30s	$\underline{53.18} \pm 3.83$	$\underline{47.48} \pm 2.34$	$\mathbf{27.46} \pm 1.08$	$\underline{42.71} \pm 1.55$
	Long	$\underline{48.88} \pm 2.35$	$\underline{16.07} \pm 1.04$	$\mathbf{17.82} \pm 0.76$	$\underline{27.59} \pm 0.90$

Spatially grounded conditioning is associated with stronger action alignment. GE-Sim 2.0 achieves the highest interaction spatial alignment scores on both robot embodiments: $81.40/73.49$ in the 30s/long AgiBot setting and $74.97/59.78$ on YAM. It also reaches 78.70 on the AgiBot policy-rollout distribution. One possible explanation is its use of pixel-aligned pose maps, which explicitly connect end-effector motion to image locations. However, the evaluated models also differ in pretraining, architecture, memory, and pretrained embodiment. The present comparison therefore does not establish that pixel-aligned conditioning alone causes the improvement.

Execution coherence degrades over long autoregressive rollouts. All evaluated models decline as the rollout horizon increases, although at different rates. As shown in Table 2, the WorldOracle-SV scores of DreamDojo and Cosmos 3 decrease from 50.56 to 27.08 and from 53.99 to 26.60, respectively. GE-Sim 2.0 degrades less, retaining 63.33 from 73.86. The results show that strong short-horizon performance does not necessarily carry over to long rollouts.

Strong action alignment does not guarantee the correct physical outcome. A model can follow the robot trajectory closely while still predicting the wrong interaction outcome. In the full-episode evaluation in Table 2, GE-Sim 2.0 obtains 73.49 in interaction spatial alignment and 96.96 in multi-view action coherence, but lower scores of 60.26 in physical state correctness and 56.24 in key interaction area fidelity. This suggests that accurate action alignment and faithful physical prediction remain separate challenges. In particular, a rollout may track the robot motion well while differing from the paired real execution in grasp state, contact, object motion, or task outcome.

5 Conclusion

We introduce WorldOracle, a benchmark for evaluating whether action-conditioned world models reproduce the physical consequences observed in the real world, given the same initial observations and action trajectories. Four metrics evaluate physical outcomes, spatial action alignment, interaction-region fidelity, and multi-view action coherence. Across seven ACWMs, we find that visually plausible rollouts can misrepresent object dynamics and interaction outcomes, even when robot-arm motion is accurately reproduced. Strong short-horizon performance also does not consistently translate into reliable long-horizon prediction. Human studies and controlled analyses support the validity and complementary roles of the metrics. Improving ACWMs with persistent action grounding and accurate modeling of object interactions remains an important challenge. We hope WorldOracle provides a useful benchmark for measuring progress toward world models that better match what actually happens in the physical world.

AI USE STATEMENT

We used generative AI tools to improve the clarity and grammar of the writing. GPT-5.5 was also used in the physical criteria annotation pipeline to generate evaluation criteria from real-world reference keyframes, as described in Sec. 3.1 and App. F. The generated criteria were independently checked and verified by human annotators against the reference images. We reviewed all AI-assisted work. Generative AI was not used to collect or modify the real-world robot execution data. We take full responsibility for the final content, claims, and artifacts presented in this work.

REPRODUCIBILITY STATEMENT

We provide supplementary code implementing all WorldOracle metrics, evaluation protocols, and result aggregation. The main paper defines the metrics, while the appendix documents dataset construction (App. E), model training and implementation (App. H), and additional metrics implementation details (App. G).

REFERENCES

Niket Agarwal, Arslan Ali, Jon Allen, Martin Antolini, Adeline Aubame, Alisson Azzolini, Junjie Bai, Maciej Bala, Yogesh Balaji, Josh Bapst, et al. Cosmos 3: Omnimodal world models for physical ai. *arXiv preprint arXiv:2606.02800*, 2026.

Arthur Allshire, Himanshu Gaurav Singh, Ritvik Singh, Adam Rashid, Hongsuk Choi, David McAllister, Justin Yu, Yiyuan Chen, Huang Huang, Pieter Abbeel, et al. Scalable behavior cloning with open data, training, and evaluation. *arXiv preprint arXiv:2606.27375*, 2026.

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, et al. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

Qingwen Bu, Jisong Cai, Li Chen, Xiuqi Cui, Yan Ding, Siyuan Feng, Shenyuan Gao, Xindong He, Xuan Hu, Xu Huang, et al. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. *arXiv preprint arXiv:2503.06669*, 2025.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris Coll-Vinent, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, et al. Sam 3: Segment anything with concepts. In *International Conference on Learning Representations*, volume 2026, pp. 138846–138923, 2026.

Yuzhi Chen, Ronghan Chen, Dongjie Huo, Yandan Yang, Dekang Qi, Haoyun Liu, Tong Lin, Shuang Zeng, Junjin Xiao, Xinyuan Chang, et al. Abot-physworld: Interactive world foundation model for robotic manipulation with physics alignment. *arXiv preprint arXiv:2603.23376*, 2026.

Cheng Chi, Zhenjia Xu, Chuer Pan, Eric Cousineau, Benjamin Burchfiel, Siyuan Feng, Russ Tedrake, and Shuran Song. Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots. *arXiv preprint arXiv:2402.10329*, 2024.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

Yufan Deng, Zilin Pan, Hongyu Zhang, Xiaojie Li, Ruoqing Hu, Yufei Ding, Yiming Zou, Yan Zeng, and Daquan Zhou. Rethinking video generation model for the embodied world. In *Forty-third International Conference on Machine Learning*, 2026.

Haoyi Duan, Hong-Xing Yu, Sirui Chen, Li Fei-Fei, and Jiajun Wu. Worldscore: A unified evaluation benchmark for world generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 27713–27724, 2025.

Chun-Kai Fan, Xiaowei Chi, Xiaozhu Ju, Hao Li, Yong Bao, Yu-Kai Wang, Lizhang Chen, Zhiyuan Jiang, Kuangzhi Ge, Ying Li, et al. Wow, wo, val! a comprehensive embodied world model evaluation turing test. *arXiv preprint arXiv:2601.04137*, 2026.

Shenyuan Gao, William Liang, Kaiyuan Zheng, Ayaan Malik, Seonghyeon Ye, Sihyun Yu, Wei-Cheng Tseng, Yuzhu Dong, Kaichun Mo, Chen-Hsuan Lin, Qianli Ma, Seungjun Nah, Loic Magne, Jiannan Xiang, Yuqi Xie, Ruijie Zheng, Dantong Niu, You Liang Tan, K. R. Zentner, George Kurian, Suneel Indupuru, Pooya Jannaty, Jinwei Gu, Jun Zhang, Jitendra Malik, Pieter Abbeel, Ming-Yu Liu, Yuke Zhu, Joel Jang, and Linxi Fan. Dreamdojo: A generalist robot world model from large-scale human videos. *arXiv preprint arXiv:2602.06949*, 2026.

Google DeepMind. Veo 3: A text-to-video generation system. Technical report, Google DeepMind, 2025. URL https://storage.googleapis.com/deepmind-media/veo/Veo-3-Tech-Report.pdf.

Yanjiang Guo, Lucy Xiaoyang Shi, Jianyu Chen, and Chelsea Finn. Ctrl-world: A controllable generative world model for robot manipulation. *arXiv preprint arXiv:2510.10125*, 2025.

Quan Huynh-Thu and Mohammed Ghanbari. Scope of validity of psnr in image/video quality assessment. *Electronics letters*, 44(13):1, 2008.

Feng Jiang, Yang Chen, Kyle Xu, Yuchen Liu, Haifeng Wang, Zhenhao Shen, Jasper Lu, Shengze Huang, Yuanfei Wang, Chen Xie, et al. Robowm-bench: A benchmark for evaluating world models in robotic manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4455–4460, 2026.

Yuxin Jiang, Shengcong Chen, Siyuan Huang, Liliang Chen, Pengfei Zhou, Yue Liao, Xindong He, Chiming Liu, Hongsheng Li, Maoqing Yao, and Guanghui Ren. Enerverse-ac: Envisioning embodied environments with action condition. *arXiv preprint arXiv:2505.09723*, 2025.

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. *arXiv preprint arXiv:2403.12945*, 2024.

Dacheng Li, Yunhao Fang, Yukang Chen, Shuo Yang, Shiyi Cao, Justin Wong, Michael Luo, Xiaolong Wang, Hongxu Yin, Joseph Gonzalez, et al. Worldmodelbench: Judging video generation models as world models. *Advances in Neural Information Processing Systems*, 38, 2026.

Soroush Nasiriany, Sep Nasiriany, Abhiram Maddukuri, and Yuke Zhu. Robocasa365: A large-scale simulation framework for training and benchmarking generalist robots. In *International Conference on Learning Representations*, volume 2026, pp. 98643–98667, 2026.

OpenAI. Introducing GPT-5.5. https://openai.com/index/introducing-gpt-5-5/, April 2026. Accessed: 2026-09-18.

OpenAI, Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. https://openai.com/research/video-generation-models-as-world-simulators, 2024.

Physical Intelligence, 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. *arXiv preprint arXiv:2504.16054*, 2025.

Boxiang Qiu, Liliang Chen, Yue Liao, Nan Wang, Lintao Wang, Jiayi Luo, Wenzhi Zhao, Shengcong Chen, Di Chen, Ye Li, et al. Ge-sim 2.0: A roadmap towards comprehensive closed-loop video world simulators for robotic manipulation. *arXiv preprint arXiv:2605.27491*, 2026.

Yu Shang, Zhuohang Li, Yiding Ma, Weikang Su, Xin Jin, Ziyou Wang, Lei Jin, Xin Zhang, Yinzhou Tang, Haisheng Su, Chen Gao, Wei Wu, Xihui Liu, Dhruv Shah, Zhaoxiang Zhang, Zhibo Chen, Jun Zhu, Yonghong Tian, Tat-Seng Chua, Wenwu Zhu, and Yong Li. Worldarena: A unified

benchmark for evaluating perception and functional utility of embodied world models. *arXiv preprint arXiv:2602.08971*, 2026a.

Yu Shang, Yinzhou Tang, Yiding Ma, Zhuohang Li, Lei Jin, Weikang Su, Xin Jin, Zhaolu Wang, Ziyou Wang, Xin Zhang, et al. Worldarena 2.0: Extending embodied world model benchmarking on modality, functionality and platform. *arXiv preprint arXiv:2605.17912*, 2026b.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, and Michaël Ramamonjisoa. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Wei-Cheng Tseng, Gashon Hussein, Yuzhu Dong, Allen Z Ren, Lucy X Shi, XuDong Wang, Sergey Levine, Zhaoshuo Li, Jinwei Gu, Florian Shkurti, et al. Sc3-eval: Evaluating robot foundation models via self-consistent video generation. *arXiv preprint arXiv:2606.18610*, 2026.

Homer Rich Walke, Kevin Black, Tony Z Zhao, Quan Vuong, Chongyi Zheng, Philippe Hansen-Estruch, Andre Wang He, Vivek Myers, Moo Jin Kim, Max Du, et al. Bridgedata v2: A dataset for robot learning at scale. In *Conference on robot learning*, pp. 1723–1736. PMLR, 2023.

Team Wan, 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 preprint arXiv:2503.20314*, 2025.

Jianyuan Wang, Minghao Chen, Shangzhan Zhang, Nikita Karaev, Johannes Schönberger, Patrick Labatut, Piotr Bojanowski, David Novotny, Andrea Vedaldi, and Christian Rupprecht. Vggt-Ω. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Zhuoyuan Wu and Jun Gao. Oscar: Omni-embodiment action-conditioned world model for robotics. *arXiv preprint arXiv:2606.04463*, 2026.

Mutian Xu, Tianbao Zhang, Tianqi Liu, Zhaoxi Chen, Xiaoguang Han, and Ziwei Liu. Kinema4d: Kinematic 4d world modeling for spatiotemporal embodied simulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4495–4500, 2026.

Xiuyu Yang, Bohan Li, Shaocong Xu, Nan Wang, Chongjie Ye, Zhaoxi Chen, Minghan Qin, Yikang Ding, Zheng Zhu, Xin Jin, et al. Orv: 4d occupancy-centric robot video generation. *arXiv preprint arXiv:2506.03079*, 2025a.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations*, volume 2025, pp. 83048–83077, 2025b.

Hu Yue, Siyuan Huang, Yue Liao, Shengcong Chen, Pengfei Zhou, Liliang Chen, Maoqing Yao, and Guanghui Ren. Ewmbench: Evaluating scene, motion, and semantic quality in embodied world models. *arXiv preprint arXiv:2505.09694*, 2025.

Chenyangguang Zhang, Botao Ye, Boqi Chen, Alexandros Delitzas, Fangjinhua Wang, Marc Pollefeys, and Xi Wang. Controllable egocentric video generation via occlusion-aware sparse 3d hand joints. In *European Conference on Computer Vision*, pp. 668–687. Springer, 2026.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 586–595. IEEE, 2018.

Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023.

Fengzhe Zhou, Jiannan Huang, Jialuo Li, Deva Ramanan, and Humphrey Shi. Pai-bench: A comprehensive benchmark for physical ai. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21522–21536, 2026.

Fangqi Zhu, Hongtao Wu, Song Guo, Yuxiao Liu, Chilam Cheang, and Tao Kong. Irasim: Learning interactive real-robot action simulators. *arXiv preprint arXiv:2406.14540*, 2024.

Appendix

This appendix is organized into 9 sections. We first present extended results and analyses of model performance (Sec. A), followed by validation of WorldOracle metrics (Sec. B). We then describe the selection of the short-horizon boundary (Sec. C), as well as the benchmark design (Sec. D, Sec. E, and Sec. F). We then describe the details of the evaluation metrics (Sec. G), before providing training and implementation details (Sec. H) and limitations (Sec. I).

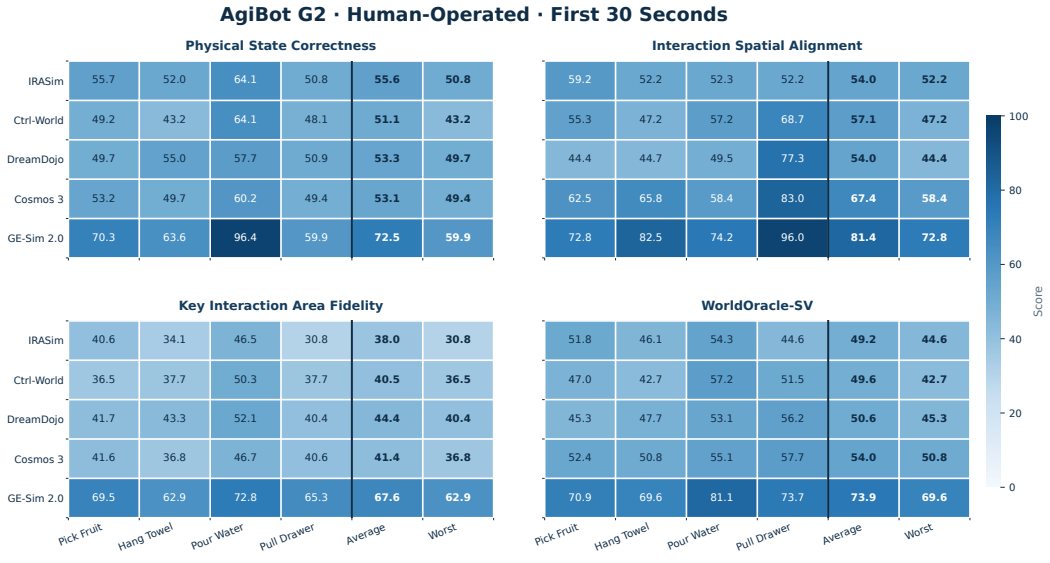

Figure S1: **Task-wise interaction-dynamics analysis on human-operated AgiBot G2 split.** Results are evaluated over the first 30 seconds on physical state correctness, interaction spatial alignment, key interaction area fidelity, and WorldOracle-SV. In each heatmap, *Average* denotes the arithmetic mean across tasks, while *Worst* denotes the lowest task score.

A Extended Results and Analyses

This section provides detailed result breakdowns across tasks, action distributions, embodiments, and evaluation horizons. The models differ in architecture, action representation, training data, and training embodiment. Moreover, adaptation is not matched across the three embodiments: AgiBot G2 uses benchmark-specific training data, YAM has a larger train–test gap, and the human-hand models are evaluated zero-shot. **The following results therefore describe performance patterns and hypotheses rather than controlled causal effects.**

A.1 Analysis across Interaction Dynamics

We further decompose model performance across tasks with different interaction dynamics. We evaluate the first 30 seconds for the human-operated distribution and complete episodes for the policy-rollout distribution. Takeaways are described below.

Pixel-aligned conditioning is associated with stronger action following. Fig. S1 shows that GE-Sim 2.0 maintains a clear advantage across all 4 AgiBot tasks. Its average WorldOracle-SV score reaches 73.9, compared with 54.0 for the strongest alternative. The advantage is particularly large in interaction spatial alignment (81.4) and key interaction area fidelity (67.6). One possible explanation is its pixel-aligned action conditioning mechanism, which provides an explicit correspondence between the specified action and the image regions in which the interaction should happen.

The task-level results also reveal differences among interaction dynamics. Models perform generally well on *pour water* in terms of physical state correctness, key interaction area fidelity, and WorldOracle-SV, and yield higher interaction spatial alignment scores on *pull drawer*. Although

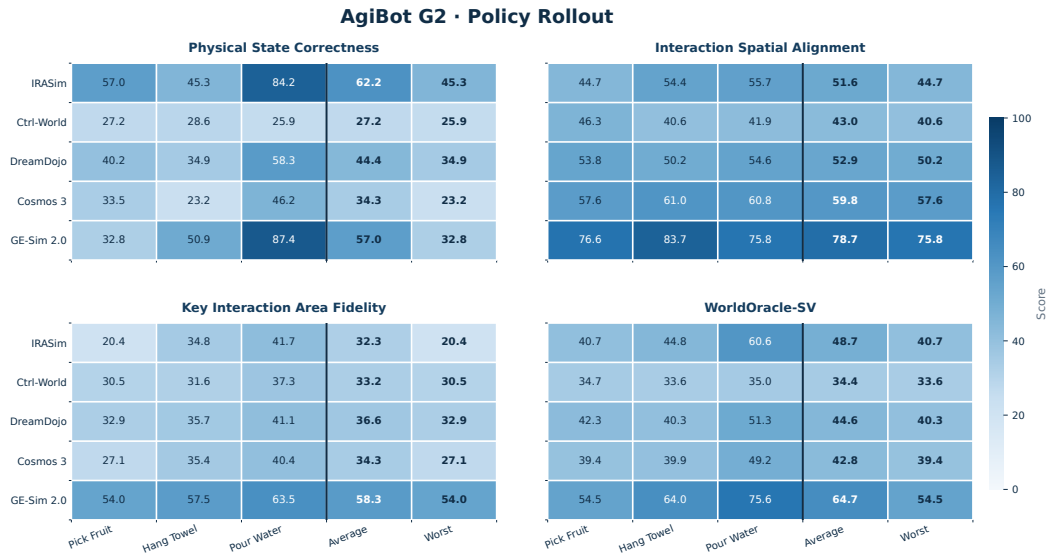

Figure S2: **Task-wise interaction-dynamics analysis on AgiBot G2 policy-rollout split.** We report results on physical state correctness, interaction spatial alignment, key interaction area fidelity, and WorldOracle-SV. In each heatmap, *Average* denotes the arithmetic mean across tasks, while *Worst* denotes the lowest task score.

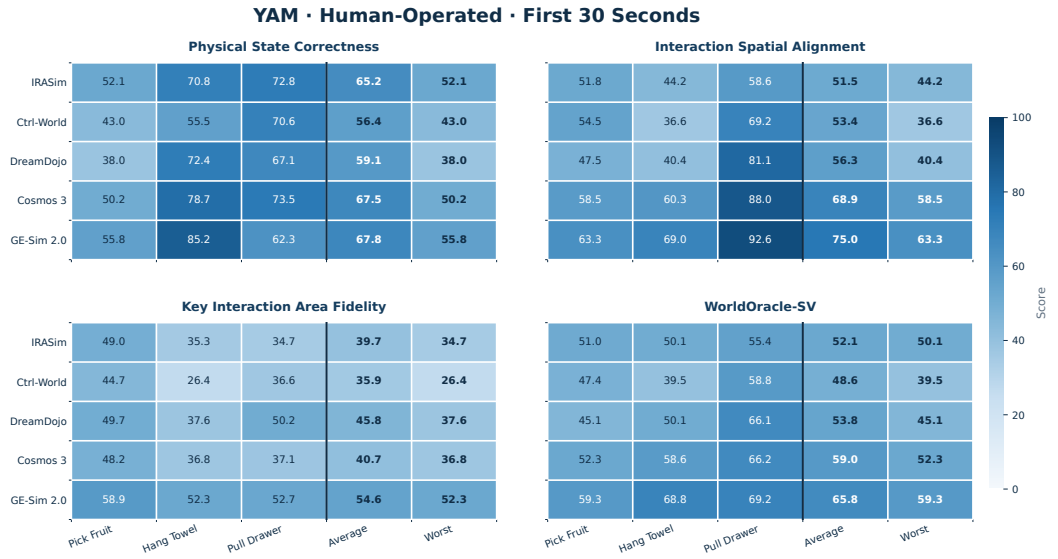

Figure S3: **Task-wise interaction-dynamics analysis on human-operated YAM split.** Results are evaluated over the first 30 seconds on physical state correctness, interaction spatial alignment, key interaction area fidelity, and WorldOracle-SV. In each heatmap, *Average* denotes the arithmetic mean across tasks, while *Worst* denotes the lowest task score.

liquid transfer is intrinsically challenging, the relatively large cup and contact region may simplify grasp localization compared with tasks involving smaller objects or more precise contact. Similarly, while articulated-object dynamics introduce additional physical complexity, the visually salient drawer and its constrained motion may make the interaction region easier to localize.

Accurate action following does not guarantee correct physical interactions. The results in Fig. S2 reveal a distinction between following the commanded trajectory and producing the intended physical outcome. IRASim achieves the highest average physical state correctness score (62.2),

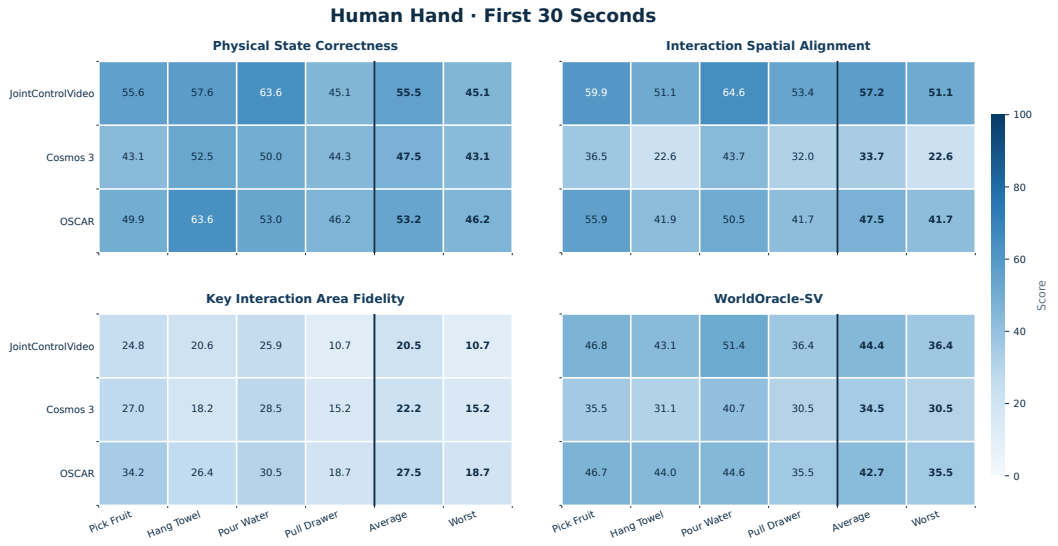

Figure S4: **Task-wise interaction-dynamics analysis for human-hand split.** Results are evaluated over the first 30 seconds on physical state correctness, interaction spatial alignment, key interaction area fidelity, and WorldOracle-SV. In each heatmap, *Average* denotes the arithmetic mean across tasks, while *Worst* denotes the lowest task score.

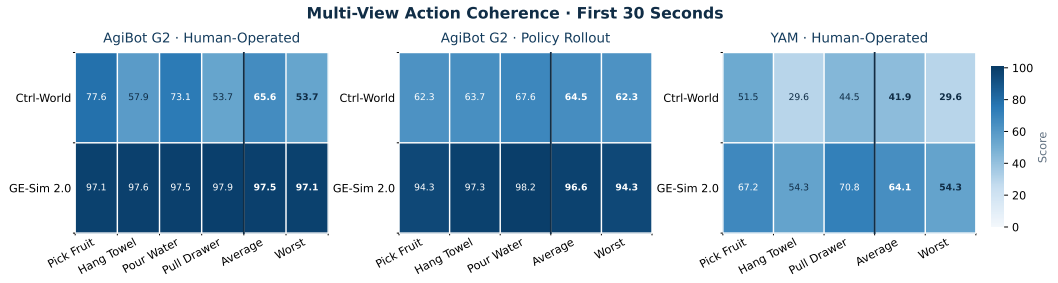

Figure S5: **Task-wise interaction-dynamics analysis for multi-view action coherence.** Results are evaluated over the first 30 seconds. In each heatmap, *Average* denotes the arithmetic mean across tasks, while *Worst* denotes the lowest task score.

slightly exceeding GE-Sim 2.0 (57.0). In contrast, GE-Sim 2.0 substantially outperforms IRASim in interaction spatial alignment (78.7 versus 51.6). This pattern is consistent with pixel-aligned action maps providing useful spatial grounding for robot-trajectory following. However, reaching the correct interaction region does not ensure that the physical interaction is coherent with the real world.

We also find that *pick fruit* is the weakest task for GE-Sim 2.0 and other models. This suggests that precise interaction with small objects remains challenging even under pixel-aligned control.

The advantage narrows across embodiments. Fig. S3 shows that GE-Sim 2.0 remains the strongest model after transfer to the YAM embodiment, achieving an average WorldOracle-SV score of 65.8. However, its margin over the strongest alternative decreases from 19.9 points on AgiBot G2 (Fig. S1) to 6.8 points on YAM. This reduction is consistent with the domain shift. The GE-Sim 2.0 backbone is pretrained on AgiBot G1, whose morphology and visual appearance are closer to those of AgiBot G2 than YAM. Nevertheless, GE-Sim 2.0 retains the highest interaction spatial alignment (75.0) and key interaction area fidelity (54.6) scores, suggesting that pixel-aligned action conditioning might be beneficial across embodiments. The narrower advantage indicates that action grounding alone does not fully resolve embodiment shift and motivates broader embodiment pretraining or explicit cross-embodiment adaptation.

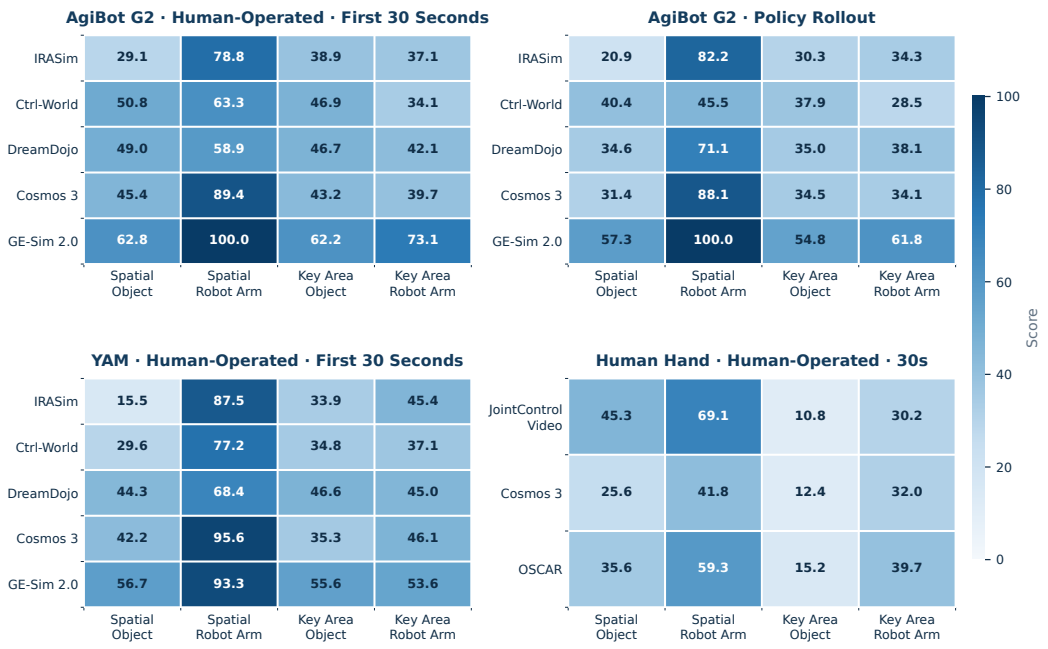

Figure S6: **Entity-centric error decomposition.** interaction spatial alignment and key interaction area fidelity are decomposed into scores computed over the object and robot-arm/hand masks, separately. In the figure, *Spatial* denotes interaction spatial alignment, while *Key Area* denotes key interaction area fidelity. Each cell reports the score for the corresponding metric and mask category.

Human-hand models trade off action following and local fidelity. Fig. S4 reveals different failure modes among the human-hand models. JointControlVideo achieves the highest average interaction spatial alignment score (57.2), indicating comparatively strong action following, but obtains the lowest key interaction area fidelity score (20.5), suggesting poor visual fidelity within the interaction regions. Cosmos 3 exhibits the opposite tendency. Its key interaction area fidelity score (22.2) slightly exceeds that of JointControlVideo, but its substantially lower interaction spatial alignment score (33.7) indicates weak correspondence between the conditioning signal and the generated interaction. OSCAR provides the best balance, combining a relatively strong interaction spatial alignment score (47.5) with the highest key interaction area fidelity score (27.5).

Multi-view coherence degrades under domain shift. Fig. S5 shows that both models perform substantially worse on YAM than on AgiBot G2. GE-Sim 2.0 decreases from an average multi-view action coherence of 97.5 on human-operated AgiBot trajectories to 64.1 on YAM, while Ctrl-World decreases from 65.6 to 41.9. This degradation likely reflects the substantial domain gap between the ABC-130K training data and the YAM evaluation distribution, including differences in tasks, scene settings, and camera observations. These results indicate that maintaining synchronized multi-view predictions remains challenging under cross-embodiment transfer, especially when there is a huge domain gap.

A.2 Entity-Centric Error Decomposition

We decompose interaction spatial alignment and key interaction area fidelity scores by the manipulated object and the robot arm (or human hand). We report results over the first 30 seconds for the human-operated distribution and over the complete episode for the policy-rollout distribution. Takeaways are described below.

Object dynamics are harder than arm motion. The interaction spatial alignment scores in Fig. S6 reveal a consistent gap between robot-arm and object alignment. For example, on human-operated AgiBot trajectories, IRASim scores 78.8 on the arm but only 29.1 on the object, while Cosmos 3 scores 89.4 and 45.4, respectively. This pattern holds for all the evaluated models across all the

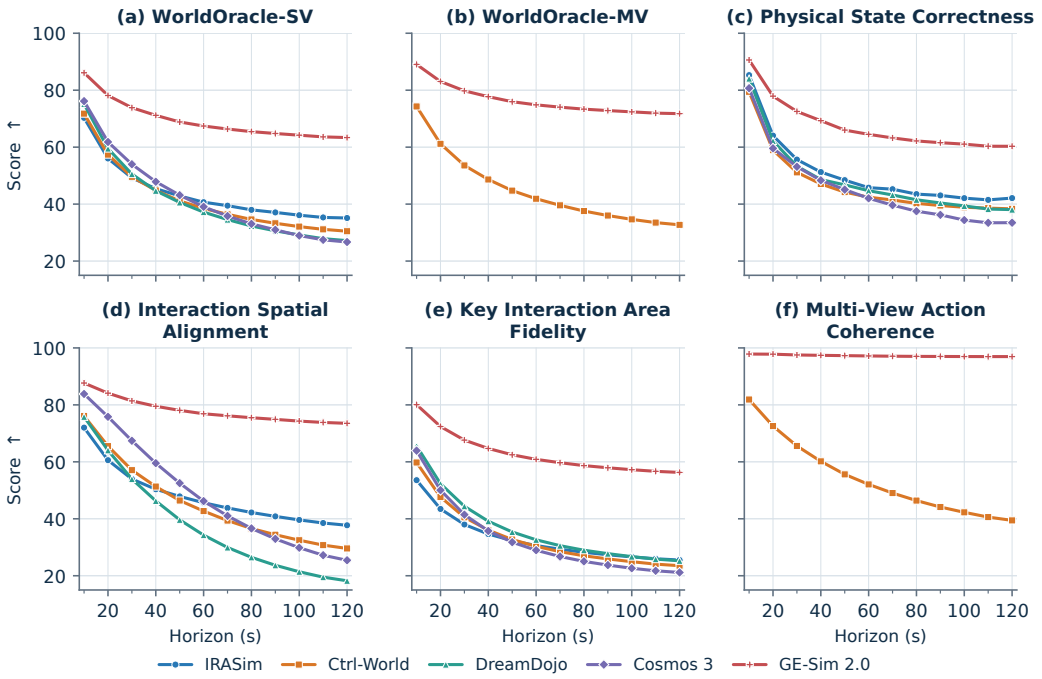

Figure S7: **Performance across cumulative rollout horizons on the human-operated AgiBot G2 distribution.** Each point reports the corresponding metric evaluated over the cumulative interval $[0, T)$, for horizons T ranging from 10 to 120 seconds. Higher scores indicate better performance.

embodiments and data distributions. This indicates that a model may accurately follow or render the specified action sequence without reproducing the plausible object motion. Modeling physical interactions and outcomes that are consistent with the real world remains a central challenge.

Human-hand interactions exhibit weaker interaction grounding. Compared with the robot embodiments, all human-hand models achieve substantially lower scores, particularly for key interaction area fidelity. Even the strongest human-hand model reaches only 15.2 on object fidelity and 39.7 on hand fidelity, while the best robot models obtain considerably higher scores. Human hands introduce highly articulated motion, deformable appearance, and fine-grained contact, which may make action grounding and local interaction rendering more difficult than in the robot settings. Accurately modeling hand–object interactions therefore remains especially challenging for current world models.

A.3 Temporal Degradation across Cumulative Rollout Horizons

We evaluate temporal degradation by computing each metric over cumulative intervals $[0, T)$, where the rollout horizon T increases from 10 to 120 seconds. A robust world model should retain high scores as autoregressive generation extends to longer horizons. Based on results shown in Fig. S7, Fig. S8, and Fig. S9, we summarize the key observations below.

AgiBot G2. Fig. S7 shows that GE-Sim 2.0 maintains a clear lead across all cumulative horizons. Its WorldOracle-SV score decreases from 86.1 at 10 seconds to 63.4 at 120 seconds, while other methods finish between 26.7 and 35.1. Among the remaining models, Cosmos 3 achieves the highest WorldOracle-SV score through 60 seconds but degrades rapidly thereafter. IRASim overtakes it at 70 seconds. This indicates that strong short-horizon performance does not necessarily imply temporal endurance.

YAM. Overall, models perform worse on YAM than on AgiBot G2, indicating greater sensitivity to the train–test domain gap. For YAM, we rely on public training data whose scenes, visual appearance, and trajectory distribution differ substantially from our evaluation set. In contrast, the AgiBot training data are collected specifically for the benchmark and therefore more closely match the corresponding

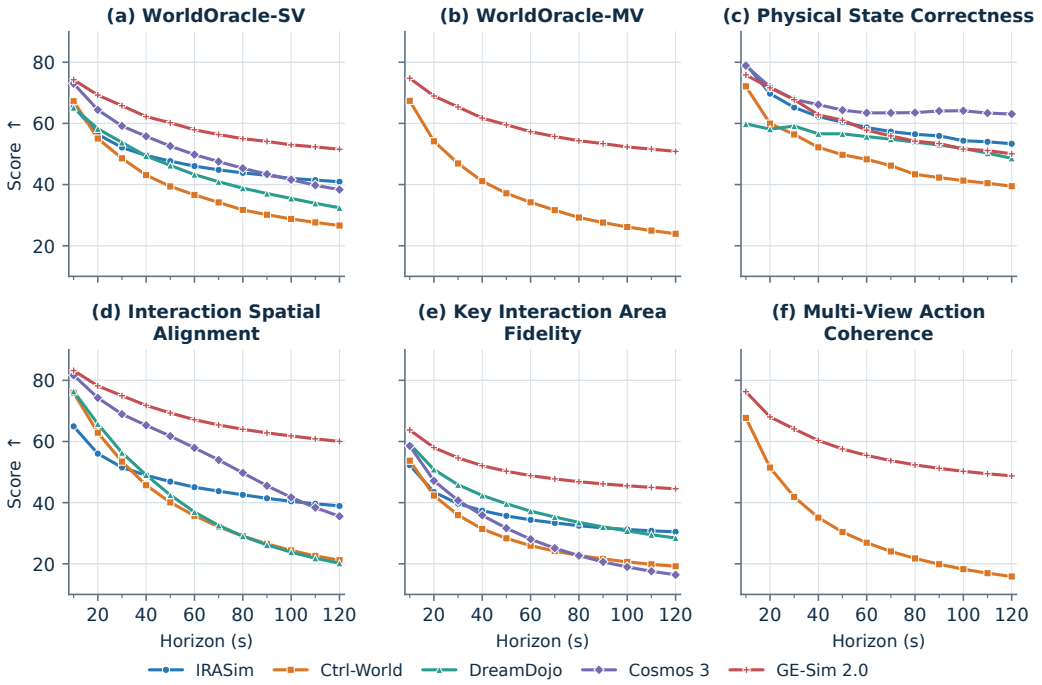

Figure S8: **Performance across cumulative rollout horizons on the human-operated YAM distribution.** Each point reports the corresponding metric evaluated over the cumulative interval $[0, T)$, for horizons T ranging from 10 to 120 seconds. Higher scores indicate better performance.

evaluation distribution. These results suggest that world-model performance decreases as the domain gap between training and evaluation increases.

GE-Sim 2.0 exhibits an additional cross-embodiment gap, particularly in physical state correctness and multi-view action coherence. Its backbone is pretrained on AgiBot G1, whose morphology and visual appearance are closer to AgiBot G2 than to YAM. Therefore, a larger embodiment shift is associated with weaker performance in this setting. This result indicates that strong performance within a closely related robot family does not necessarily transfer to a morphologically different embodiment.

Human hand. As shown in Fig. S9, human-hand models perform worse than models evaluated on the two robot embodiments. We attribute this gap to two factors. First, the models are evaluated zero-shot on the human-hand distribution. Second, human-hand interactions involve highly articulated motion, deformable appearance, and fine-grained contact, making them intrinsically more difficult to model than fixed-morphology robot interactions.

Across all three models, physical state correctness is considerably more durable than interaction spatial alignment and key interaction area fidelity. For example, JointControlVideo retains a physical state correctness score of 56.5 at 120 seconds, whereas its interaction spatial alignment and key interaction area fidelity scores decrease to 24.9 and 8.4, respectively. Thus, although current models can preserve coarse physical plausibility over long horizons, accurately grounding and rendering fine-grained hand–object interactions remains a key challenge.

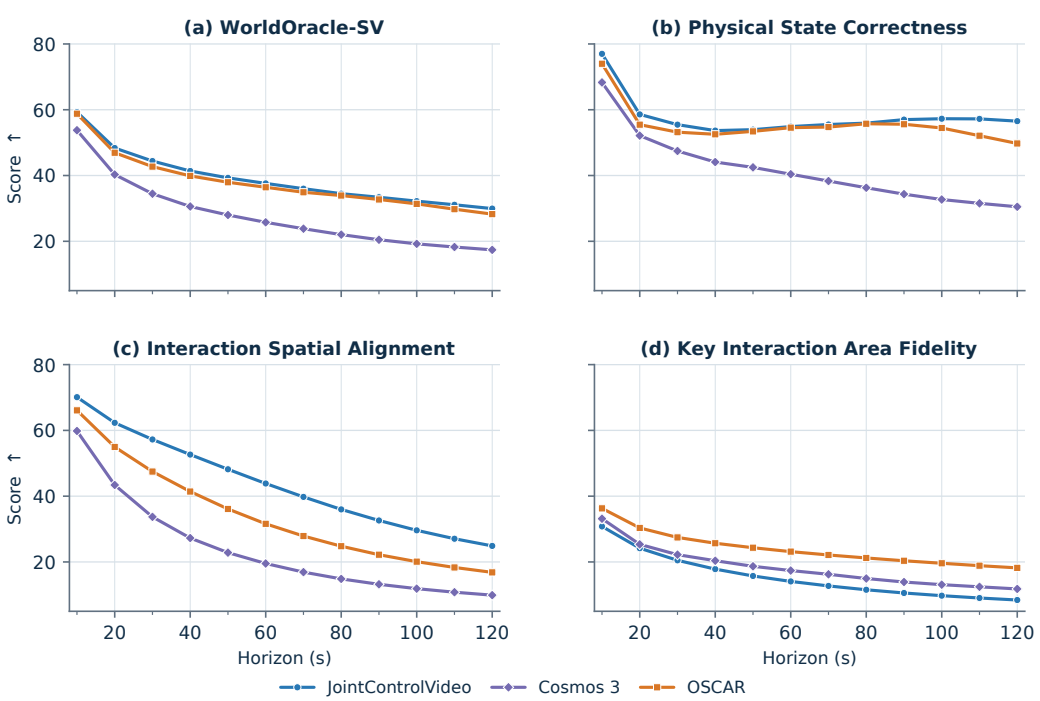

Figure S9: **Performance across cumulative rollout horizons on the human-hand distribution.** Each point reports the corresponding metric evaluated over the cumulative interval $[0, T)$, for horizons T ranging from 10 to 120 seconds. Higher scores indicate better performance.

B Validation of WorldOracle Metrics

B.1 Human–Metric Alignment Study

We conduct a human study to examine whether preferences induced by the WorldOracle metrics agree with human judgements. The study contains 128 prediction pairs spanning the AgiBot human-operated, AgiBot policy-rollout, YAM, and human-hand distributions. We collect responses from 119 participants. Each trial presents two anonymous predictions in randomized order together with the real-world reference, and participants select the prediction that better satisfies the corresponding evaluation dimension.

WorldOracle metrics align consistently with human judgments. As shown in Fig. S10, all 4 WorldOracle metrics achieve high overall agreement with human preferences, ranging from 88.1% to 93.2%. Importantly, strong agreement is observed across different metric score-gap groups, indicating that the alignment is not driven only by a small number of easily distinguishable prediction pairs. Even within the group with small score gap, agreement remains substantially above chance for all 4 evaluation dimensions.

The strong pair-level Spearman correlations ($\rho = 0.72$–0.82) further show that larger metric differences correspond to more decisive human preferences. Together, these results demonstrate that WorldOracle metrics align with human preference across both difficult and clearly distinguishable comparisons.

B.2 Metric Case Studies

Outcome correctness and action alignment are complementary. Fig. S11 illustrates both directions of the distinction between physical state correctness and interaction spatial alignment. In the pull-drawer example, the predicted arm motion remains well aligned with the real execution (interaction

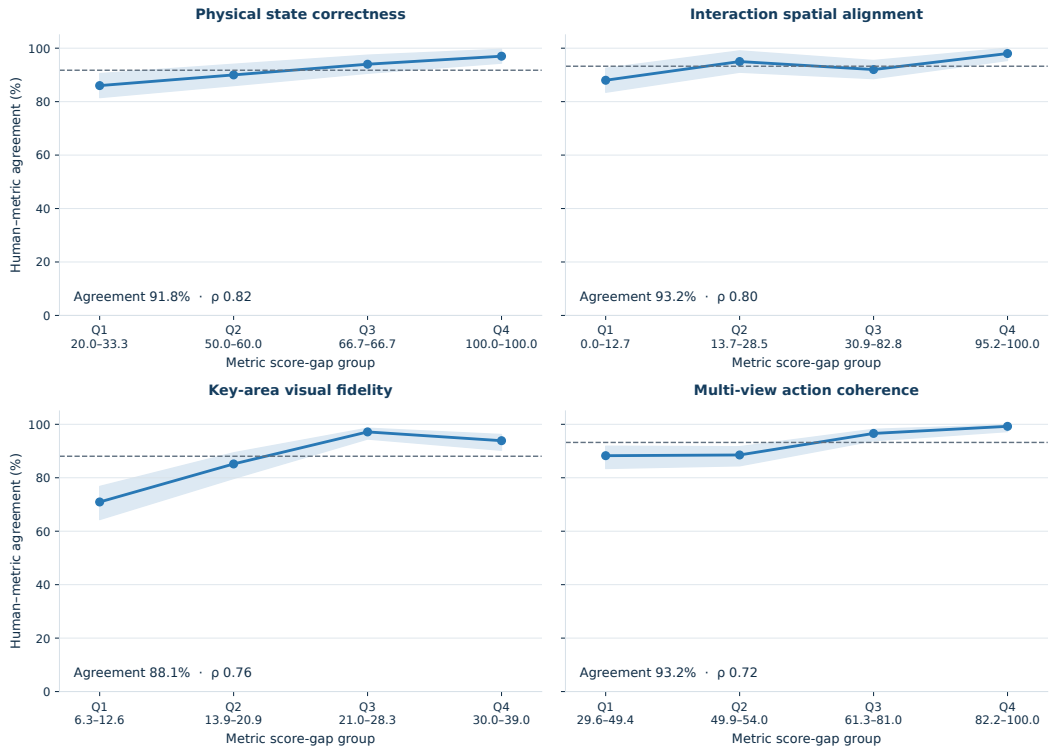

Figure S10: **Alignment between WorldOracle metrics and human judgments.** Each panel reports the fraction of human choices that agree with the preference induced by the corresponding metric, grouped by metric score gap. Shaded regions denote 95% Wilson confidence intervals, and dashed horizontal lines indicate overall agreement. The reported Spearman coefficient ρ measures the pair-level association between the signed metric-score difference and the human preference rate.

(a) Aligned action, incorrect physical outcome



Criterion: all drawers are closed.
Physical: 0.0 Spatial: 80.8

(b) Correct physical outcome, misaligned action



Criterion: the yellow towel is grasped by the gripper.
Physical: 100.0 Spatial: 9.7

Figure S11: **Complementary failure modes captured by physical state correctness and interaction spatial alignment.** Each panel compares the real and predicted images at the annotated outcome frame. *Physical* is the criterion-level physical state correctness score at this frame, while *Spatial* is the interaction spatial alignment score computed over a four-second local window centered on it. In (a), the prediction approximately follows the robot motion but incorrectly leaves the green drawer open when the real execution closes all drawers. In (b), the model predicts that the yellow towel is grasped, satisfying the stated physical criterion, but renders the arm and towel along a trajectory that differs from the real execution.

spatial alignment $= 80.8$), but the generated drawer state contradicts the annotated outcome that all drawers are closed (physical state correctness $= 0.0$). Conversely, in the hang-towel example, the prediction satisfies the criterion that the yellow towel is grasped (physical state correctness $= 100.0$), while the generated arm and towel trajectory is poorly aligned with the real execution (interaction spatial alignment $= 9.7$). Thus, neither metric subsumes the other. Spatial action following does not

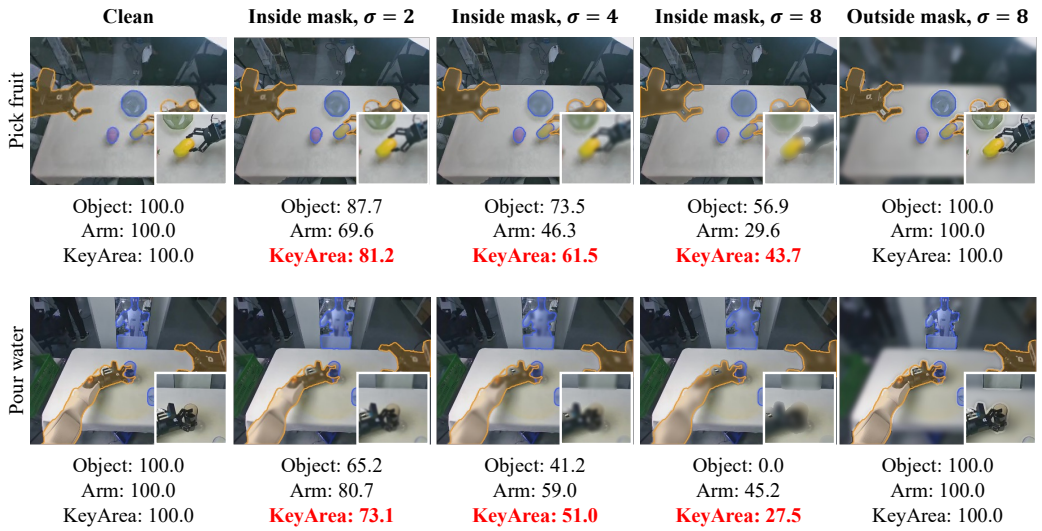

Figure S12: **Controlled case study of key interaction area fidelity.** Starting from clean real interaction frames, we apply Gaussian blur either within the object and robot-arm masks ($\sigma \in \{2, 4, 8\}$) or only outside this region ($\sigma = 8$). Blue and orange contours denote the object and robot-arm masks, respectively. Each panel reports the object score, arm score, and their aggregate key interaction area fidelity score. Increasing mask-local blur produces a monotonic decrease in key interaction area fidelity for both an object-centric Pick Fruit example and a contact-centric Pour Water example, whereas applying the strongest blur only to the background leaves the score unchanged.

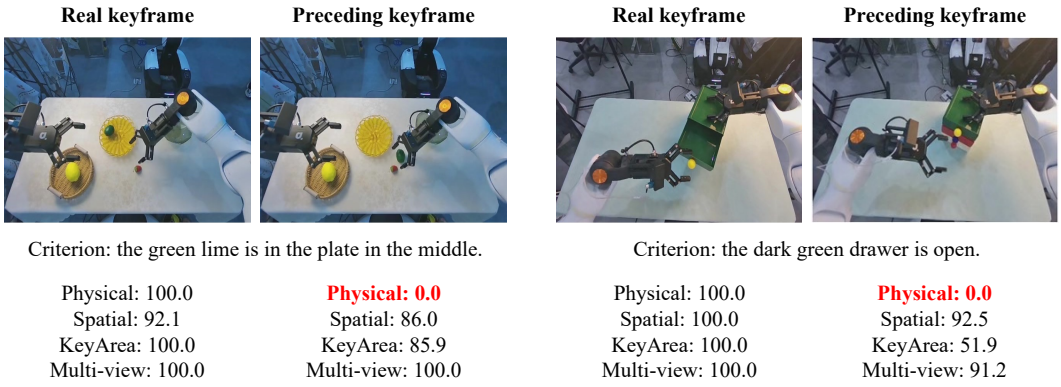

Figure S13: **Qualitative examples of the outcome-swap intervention.** Although both frames come from the same episode, the preceding keyframe contradicts the newly introduced physical-state criterion. Each example reports all four metric scores.

guarantee the correct physical consequence, and matching an outcome criterion does not guarantee a faithful execution trajectory.

Key-area fidelity is sensitive to local interaction quality. Fig. S12 isolates the spatial locality of key interaction area fidelity using controlled corruptions of two clean reference frames. For Pick Fruit, the aggregate score decreases from 100.0 on the clean frame to 81.2, 61.5, and 43.7 as the mask-local blur increases from $\sigma = 2$ to 4 and 8. The Pour Water example exhibits the same monotonic trend, decreasing from 100.0 to 73.1, 51.0, and 27.5. In contrast, severe background-only blur leaves both examples at 100.0 because the evaluated interaction pixels remain unchanged. These cases show that key interaction area fidelity responds to progressively degraded object and contact appearance while remaining localized to the interaction regions it is designed to evaluate.

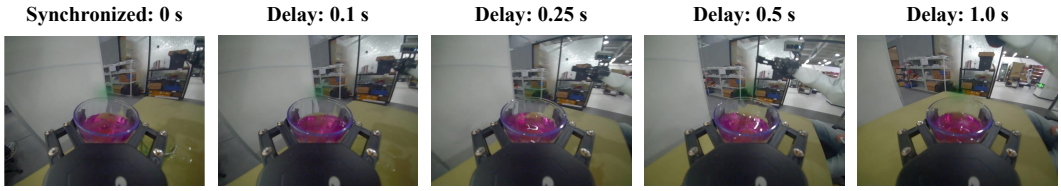

Figure S14: **Cross-view desynchronization across increasing delays.** We progressively delay only the left auxiliary view while retaining the top-head observation and right view at time t. All four scores are shown for each condition. In this example, multi-view action coherence decreases monotonically from 100.0 to 94.8, 83.7, 82.4, and 81.3 as the delay increases, whereas the three top-head metrics remain unchanged.

B.3 Controlled Metric Selectivity

Outcome swapping primarily affects physical state correctness. We construct a balanced event-level set containing ten state transitions from each task. To ensure that the intervention tests an actual state change, we retain only keyframes whose physical criterion differs from the preceding keyframe and evaluate the newly introduced criterion. The perturbed sample replaces the original keyframe with the preceding keyframe from the same episode, while retaining the original outcome keyframe as the reference. Fig. S13 shows two representative interventions: the swapped keyframe depicts a similar scene layout and robot trajectory but different fruit placement or drawer-state transition.

Across the 40 events, we compute the relative decrease as $100(S_{\text{clean}} - S_{\text{swap}})/S_{\text{clean}}$. The intervention reduces physical state correctness by 67.9% on average, substantially more than key interaction area fidelity (34.3%), interaction spatial alignment (20.0%), or multi-view action coherence (12.3%). Replacing an outcome keyframe also changes local object appearance and robot–object geometry. Nevertheless, the largest response occurs in the intended physical state correctness metric, supporting its sensitivity to incorrect interaction outcomes.

Cross-view desynchronization. We test whether multi-view action coherence detects temporal disagreement between camera streams by delaying exactly one auxiliary view while keeping the top-head view and the other auxiliary view synchronized. We independently perturb the left and right auxiliary views in 40 interaction events and average their effects. As the delay increases from 0.1 to 0.25, 0.5, and 1.0 seconds, the mean relative decrease in multi-view action coherence grows monotonically from 0.7% to 1.7%, 2.9%, and 4.4%, respectively. At the medium 0.5-second delay, the decrease is 2.9%. Because physical state correctness, interaction spatial alignment, and key interaction area fidelity use only the unchanged top-head input in this intervention, their scores remain exactly unchanged. This controlled result isolates the intended sensitivity of multi-view action coherence while demonstrating invariance of the single-view metrics to an off-target camera perturbation. Qualitative examples are shown in Fig. S14.

B.4 Metric Robustness and Stability

B.4.1 Nuisance Robustness and Model-Ranking Stability

We test whether each metric is robust to irrelevant perturbations while remaining sensitive to the property it is designed to measure. The study contains 40 interaction events from four tasks and seven models, yielding 280 prediction–reference pairs. We report the mean score change from the unperturbed setting and the Spearman correlation between model rankings.

Appearance robustness. We apply JPEG compression, downsample–upsample degradation, and brightness and contrast changes. As shown in the top row of Fig. S15, interaction spatial alignment and multi-view action coherence are highly stable: their mean changes remain within 0.45 and 0.70



Figure S15: **Nuisance robustness and sensitivity.** Left panels show the signed mean score change relative to the default setting; right panels show model-ranking Spearman correlation. Appearance perturbations leave geometry-centered metrics nearly unchanged while key interaction area fidelity appropriately responds to visible degradation. Mask morphology changes absolute scores more strongly but generally preserves rankings. Temporal subsampling produces negligible score changes and identical rankings. The dashed line denotes $\rho = 0.9$.

points, respectively, and their model rankings are unchanged ($\rho = 1.00$). Physical state correctness also changes by at most 1.9 points on average, although its ranking stability is lower ($\rho = 0.75$–0.89). In contrast, key interaction area fidelity appropriately responds to visible degradation: $0.5\times$ resizing reduces its score by 3.8 points, while $\pm 20\%$ brightness changes reduce it by 8.0–9.5 points. Its rankings remain largely stable ($\rho = 0.86$–1.00).

Mask robustness. We perturb object and robot masks through erosion, dilation, and translation. As shown in the middle row of Fig. S15, translations of 5 pixels change the metric means by at most 0.89 points, while 10-pixel translations change them by at most 2.15 points. Erosion and dilation have larger effects because they directly remove interaction pixels or introduce surrounding context. Despite these absolute score changes, model rankings remain stable for interaction spatial alignment ($\rho \geq 0.96$) and key interaction area fidelity ($\rho \geq 0.89$). The lowest multi-view action coherence correlation is 0.80.

Temporal sampling. As shown in the bottom row of Fig. S15, reducing the sampling frequency to $0.25\times$ changes the mean scores by only 0.26 points for interaction spatial alignment, 0.27 for key interaction area fidelity, and 0.08 for multi-view action coherence. All model rankings remain identical to the default evaluation ($\rho = 1.00$). The benchmark conclusions are therefore insensitive to moderate temporal subsampling.

Overall findings. The results support the robustness of the proposed metrics: geometry-based metrics remain stable under appearance changes, key interaction area fidelity responds to the visual degradation it is intended to measure, and model rankings are generally preserved under mask noise and temporal subsampling.

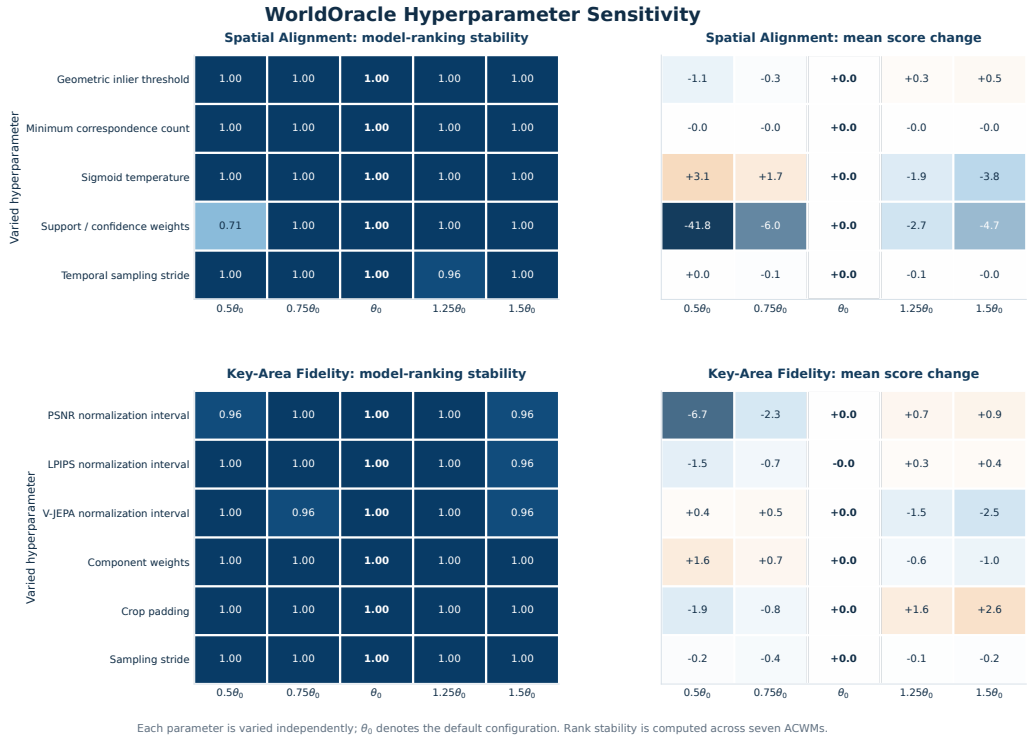

Figure S16: **Hyperparameter sensitivity of Spatial Alignment and Key-Area Fidelity.** We vary one setting at a time from $0.5\theta_0$ to $1.5\theta_0$, where θ_0 is the default. The left panels show agreement with the default ranking of seven ACWMs; the right panels show the average score change.

B.4.2 Hyperparameter and ranking stability.

We vary the main thresholds, weights, crop padding, normalization ranges, and sampling strides of both interaction spatial alignment and key interaction area fidelity metrics. The main study contains 280 model–event pairs from seven ACWMs. Sampling stride is tested on an additional dense subset of 56 pairs.

As shown in Fig. S16, the model rankings remain nearly unchanged under most settings. For interaction spatial alignment (denoted as "Spatial Alignment" in the figure), the geometric threshold, correspondence count, and sampling stride change the mean score by at most 1.1 points. The only clear failure occurs when the feature-evidence weights are halved: the mean drops by 41.8 points and rank agreement decreases to $\rho = 0.71$. At $0.75\theta_0$ and above, the original ranking is fully recovered.

Key interaction area fidelity (denoted as "Key-Area Fidelity" in the figure) is also stable, with $\rho \geq 0.96$ in every test. Its largest shift is -6.7 points when the PSNR normalization range is halved; component weights, crop padding, and sampling stride all preserve the model ranking.

C Selection of the Short-Duration Boundary

We select the short-duration boundary based on public robot-manipulation datasets. For datasets with episode-level metadata, duration is computed as the number of recorded frames divided by the nominal frame rate. When only aggregate timing information is available, we use the range reported by the corresponding paper.

As shown in Fig. S17, many atomic manipulation datasets operate primarily below 30 seconds. For example, BridgeData V2 (Walke et al., 2023) has a P50–P90 interval of 7.0–9.6 seconds, and ALOHA (Zhao et al., 2023) reports episodes lasting 8–14 seconds. PushT (Chi et al., 2025) has a P50–P90 interval of 12.2–16.8 seconds. DROID (Khazatsky et al., 2024) is broader, with a median

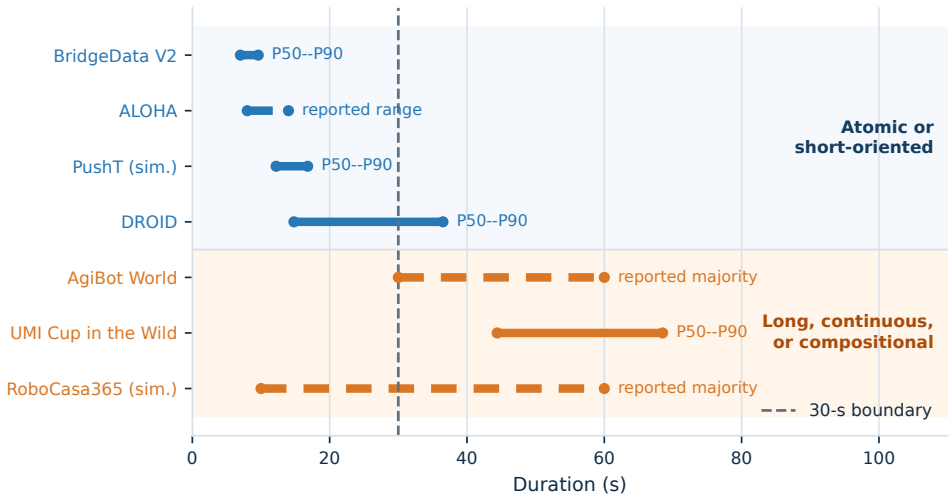

Figure S17: **Duration statistics of public robot-manipulation datasets.** Blue rows denote short, atomic datasets, whereas orange rows denote long, compositional datasets. Each row is annotated with its available timing statistic: an episode-level P50–P90 interval, or a reported majority range. Simulated datasets are marked with "sim." explicitly. The dashed line indicates our selected 30-second short-horizon boundary.

of 14.8 seconds, and 84.9% of its episodes are no longer than 30 seconds. In contrast, datasets containing longer, continuous, or compositional behavior extend beyond this boundary. AgiBot World (Bu et al., 2025) reports that most episodes last 30–60 seconds, UMI Cup (Chi et al., 2024) has a P50–P90 interval of 44.4–68.5 seconds, and most RoboCasa365 (Nasiriany et al., 2026) episodes span 10–60 seconds.

These observations place 30 seconds as an approximate transition between short, atomic behaviors and longer, compositional sequences. We therefore evaluate the first 30 seconds as a complementary short-duration regime while retaining full-episode evaluation as the long-horizon setting.

D Benchmark Task Definition

The four tasks are designed to cover distinct interaction dynamics while retaining visually verifiable outcomes.

Pick fruit. This task evaluates rigid-body manipulation. Each scene contains two to four fruits and one to three storage containers. The embodiment repeatedly transfers fruits between the tabletop and the containers. Each storage container may contain at most one piece of fruit. Successful segments include grasping a designated fruit and placing it in the intended location. Failure segments include empty grasps, drops, collisions, and fruits falling outside the reachable workspace.

Hang towel. This task evaluates deformable-object interaction. Each scene contains two or three colored towels and a drying rack. The embodiment picks up a towel, hangs it on the rack, removes it, or returns it to the tabletop. Failure outcomes include empty grasps, dropping the towel, unsuccessful placement on the rack, and the towel falling onto the table or floor.

Pour water. This task evaluates fluid interaction. Each scene contains two to four cups, at least one of which initially contains colored liquid. A typical successful segment consists of grasping a source cup, transferring its liquid into an empty cup, and returning the source cup to the table. Failures include empty grasps, dropped or tilted cups, incomplete transfer, and liquid spilling onto the tabletop. Colored liquid is used to make fluid transfer and spills more observable to the VLM evaluator. Each episode uses only one liquid color.

Pull drawer. This task evaluates articulated-object interaction. Each scene contains a multi-tier drawer placed on the tabletop. The drawer is initialized either with all tiers closed or with one tier

Table S1: **AgiBot G2 training-set statistics.** Duration is reported as mean ± standard deviation and observed minimum–maximum in seconds.

Task	Episodes	FPS	Mean ± SD (s)	Range (s)	Hours
Pick fruit	426	30	60.19 ± 1.62	56.27–67.10	7.12
Hang towel	391	30	60.57 ± 1.73	54.70–67.30	6.58
Pour water	523	30	61.82 ± 3.05	51.83–64.94	8.98
Pull drawer	439	30	60.70 ± 1.35	55.30–66.77	7.40
Total	**1,779**	**30**	$\mathbf{60.88 \pm 2.21}$	**51.83–67.30**	**30.08**

Table S2: **Evaluation-set statistics.** The evaluation suite contains 335 episodes from human-operated and policy-rollout distributions. Duration is reported as mean ± standard deviation and observed minimum–maximum in seconds.

Embodiment	Action source	Task distribution	Episodes	FPS	Mean ± SD (s)	Range (s)	Hours
AgiBot G2	Human-operated	Pick (25), Hang (25), Pour (25), Pull (25)	100	30	118.70 ± 3.88	108.27–130.47	3.30
AgiBot G2	Policy-rollout	Pick (20), Hang (20), Pour (20)	60	30	22.59 ± 13.01	4.93–60.00	0.38
YAM	Human-operated	Pick (25), Hang (25), Pull (25)	75	30	122.90 ± 4.80	110.00–138.77	2.56
Human hand	Human-operated	Pick (25), Hang (25), Pour (25), Pull (25)	100	15	125.35 ± 2.28	121.40–134.53	3.48
Total	–	Pick (95), Hang (95), Pour (70), Pull (75)	**335**	–	$\mathbf{104.41 \pm 38.85}$	**4.93–138.77**	**9.72**

open. An open drawer may contain a clearly visible object. The embodiment interacts with a drawer tier by grasping its handle and opening or closing it. A successful segment requires the intended tier to reach the open or closed state. Failures include missing the handle, and losing contact during pulling or pushing.

E Additional Details on Data Curation

Each episode contains multiple interaction segments, including both successful and failed manipulation attempts. Across episodes, we vary object identities, quantities, sizes, initial configurations, and illumination conditions.

E.1 Training Suite

The newly collected AgiBot G2 training split contains 1,779 episodes across the four benchmark tasks. Statistics are shown in Table S1. Each episode is recorded at 30 FPS and lasts approximately one minute. Within each task, the robot starts from a fixed initial pose, and only its two arms and grippers are controlled. Left- and right-arm executions are balanced during data collection. The training suite uses a controlled range of scene variations, including a restricted set of tablecloth colors, a fixed illumination color, and reduced sets of fruits, containers, cups, liquid colors, towels, and drawer configurations.

E.2 Human-Operated Distribution

We collect held-out human-operated evaluation episodes for the AgiBot G2, YAM, and human-hand embodiments. The robot trajectories are produced through teleoperation, while the human-hand trajectories are produced through direct human execution. As shown in Table S2, the AgiBot G2 and human-hand splits each contain 100 episodes, with 25 episodes per task. The YAM split contains 75 episodes, with 25 episodes each for *pick fruit*, *hang towel*, and *pull drawer*. Each episode lasts approximately two minutes and contains multiple consecutive interaction segments.

For AgiBot G2, the evaluation split expands the scene variations used during training. It includes held-out fruit instances with new colors and sizes, as well as new container types, towel colors, liquid colors, and drawer configurations. We additionally introduce unseen tablecloth colors and vary the color, angle, and position of scene illumination. The training and evaluation splits therefore share the same task semantics and retain some overlapping appearance conditions while holding out selected objects, scene configurations, and visual conditions. This design evaluates whether an ACWM generalizes learned action–outcome relationships to new visual configurations instead of reproducing memorized trajectories.

E.3 Policy-Rollout Distribution

We additionally collect a policy-rollout evaluation split on the physical AgiBot G2 robot. A $\pi_{0.5}$ policy (Physical Intelligence et al., 2025) trained using our curated training data interacts with the real robot in closed loop. We record the resulting observations and low-level actions for evaluation. This split contains 60 episodes, with 20 episodes each for *pick fruit*, *hang towel*, and *pour water*.

F Additional Details on the Annotation Pipeline

F.1 Physical Criteria Annotation and Evaluation

We formulate physical-criteria annotation and evaluation as a criteria-authoring and criteria-judging procedure. Let $v_0 = \texttt{top_head}$ denote the primary camera view. In the criteria authoring stage, GPT-5.5 (OpenAI, 2026) receives the episode's initial observation $O_{i,0}^{v_0}$ and the outcome keyframe $K_{i,n}^{v_0}$. The initial observation establishes a canonical inventory of the objects and their visual identities throughout the episode, while the outcome keyframe determines the current physical state. GPT-5.5 generates a set of visually verifiable criteria $\mathcal{C}_{i,n} = \{c_{i,n,j}\}_{j=1}^{J_{i,n}}$, where $J_{i,n}$ is the number of physical criteria of the segment n of episode i. A revision pass normalizes entity names against the inventory created from the initial observation without changing the states asserted by the criteria. During the judging stage, GPT-5.5 receives the same initial-observation reference, a predicted outcome keyframe $\hat{K}_{i,n}^{v_0}$, the canonical object inventory, and $\mathcal{C}_{i,n}$. It answers each criterion with a binary score $y_{i,n,j} \in \{0, 1\}$, where $y_{i,n,j} = 1$ indicates that criterion $c_{i,n,j}$ is visibly satisfied in the predicted outcome keyframe. We use separate task-specific prompts since the relevant entities, physical states, and sources of visual ambiguity differ substantially across interaction dynamics. The complete authoring and judging prompts are shown in the boxes below.

Pick Fruit: Physical Criteria Authoring and Judging

```
[AUTHORING SYSTEM PROMPT]

You are an expert annotator for a robot manipulation benchmark.
SETTING: A robot with two grippers (hands) performs a fruit pick-and-place task. It picks up
a fruit from a table, or a basket / fruit plate, and places it into a basket / fruit plate,
or puts it down on the table.

YOU ARE GIVEN TWO IMAGES:
- Image 1 is the episode's FIRST frame: a clear, mostly unoccluded view. Use it to establish
the GROUND TRUTH set of objects in the scene -- every fruit's type/color and every
basket/plate's color, style, and location.
- Image 2 is the KEY FRAME you must annotate.

IDENTITY RULE (very important): in the key frame a fruit or container can look different
because of occlusion, motion blur, or viewpoint -- e.g. a mango may look like a banana or an
orange. ALWAYS take the identity (type/color) of every fruit and container from Image 1. If
something in the key frame roughly matches an object from Image 1, treat it as that same
object. NEVER introduce a fruit or container that is not present in Image 1.

YOUR JOB: Write the 'success_criteria' for the KEY FRAME (Image 2) -- a list of concrete,
visually verifiable conditions that hold in Image 2.

HOW TO WRITE GOOD CRITERIA:
- Keep every criterion as short and simple as possible -- one plain statement.
- Every criterion has a FRUIT (named exactly as identified in Image 1) as its subject. Do
NOT write criteria about a basket/plate, the gripper, or anything else as the subject, and
do not describe the container's own state.
```

```
- In the key frame each fruit is in exactly ONE of these states:
    * on the table,
    * in a specific basket / fruit plate, or
    * not visible in the key frame.
- ONLY write a criterion you are VISUALLY CERTAIN about. If a fruit is too occluded (e.g.
mostly hidden by the gripper/arm) or its spatial state is otherwise ambiguous, OMIT it
entirely rather than guess.
- If a fruit that exists in Image 1 is clearly absent / not visible in the key frame, say so
explicitly, e.g. "The apple is not in the view." Use this only when you are sure it is not
visible -- NOT when its state is merely hard to judge.
- Identify the fruit by its color/type from Image 1 (e.g. "the red apple", "the banana").
- When a fruit is in a container, name which container by its color, style, and location so
different baskets/plates can be told apart. The container only locates the fruit.
- Each criterion is ONE short, self-contained English sentence.
- Write a criterion for each fruit whose state you can judge with high visual certainty. Do
not pad or guess.

GOOD EXAMPLES:
- "The red apple is in the brown wicker basket on the top left."
- "The orange is in the white plate on the right."
- "The banana is on the table."
- "The mango is not in the view."

OUTPUT CONTRACT: Respond with STRICTLY a JSON array of strings and NOTHING else. No
markdown, no code fences, no explanations, no object keys.
Example: ["...", "..."]

[AUTHORING USER PROMPT]

Image 1 is the episode's first frame (the reference for the identity of every fruit and
container). Image 2 is the key frame to annotate.

Using Image 1 to fix the identity of each fruit and container, write the success_criteria
that hold in the key frame (Image 2). Respond with ONLY a JSON array of strings.


--------------------------------------------------------------------------------

[JUDGING SYSTEM PROMPT]

You are an expert physical-status judge for a robot FRUIT pick-and-place benchmark. A robot
moves fruits between a table and various baskets / fruit plates. You are shown ONE key frame
plus a list of success criteria. Each criterion names a fruit and asserts WHERE it is: on
the table, in a specific basket / fruit plate (named by color / style / location), grasped
by the gripper (optionally hovering above the table or above a named container), or not
visible in the frame.

Your job: for each criterion, return a binary 1 (clearly satisfied) or 0 (clearly not
satisfied).

IDENTITY -- DISAMBIGUATE AGAINST THE OBJECT INVENTORY:
- You may be given TWO images: IMAGE 1 is the episode's FIRST frame and IMAGE 2 is the frame
to JUDGE.
- The user message gives the COMPLETE set of fruits and containers in the scene.
- Do NOT demand a textbook shape/type match. Match each frame object to the CLOSEST
inventory item, primarily by COLOR and then by rough shape / size.
- A criterion is satisfied when the object in the stated location best matches the
criterion's named object among the inventory.
- Score 0 when the object best matches a different inventory item or when no matching object
is present.
- Disambiguate containers by color, style, and location.

WHAT TO VERIFY:
- The fruit's location: on the table, in the named container, grasped by the gripper, or not
visible.
- For grasped / hovering criteria, verify both the grasp and, when stated, whether it is
above the table or the named container.
- For "not visible", verify that no matching inventory fruit is visible.

Reserve 0 for clear contradictions: a different fruit, wrong container, wrong grasp/hover
location, present when it should be absent, or absent when it should be present. A
shape/type mismatch alone is not a reason to fail.

Success criteria for this frame: for each, judge 1/0 whether the named fruit is in the
described place. Match the object to the closest item in the inventory.

Return exactly:
{
```

```
  "aspect_scores": [0 or 1, ...],
  "reasoning": ["brief visible evidence", "..."],
  "explanation": "one sentence summary"
}

The aspect_scores array must contain exactly one integer per criterion, in the same order.
```

Hang Towel: Physical Criteria Authoring and Judging

```
[AUTHORING SYSTEM PROMPT]

You are an expert annotator for a robot manipulation benchmark.

SETTING: A robot with a gripper picks a towel up from the table and hangs / drapes it over a
horizontal hanging ROD (a clothes-drying bar / pole), OR takes a towel off the rod and puts
it down on the table. The scene's objects are TOWELS.

YOU ARE GIVEN TWO IMAGES:
- Image 1 is the episode's FIRST frame: a clear, mostly unoccluded view. Use it to establish
the GROUND TRUTH set of towels in the scene -- every towel's COLOR (and its position, used
only to tell same-colored towels apart).
- Image 2 is the KEY FRAME you must annotate.

IDENTITY RULE: in the key frame a towel can look different because of folding, draping,
occlusion, the gripper, or viewpoint -- its shape is unreliable. ALWAYS take the identity
(COLOR) of every towel from Image 1. If a towel in the key frame roughly matches a towel
from Image 1 by color, treat it as that same towel. NEVER introduce a towel color that is
not present in Image 1.

YOUR JOB: Write the 'success_criteria' for the KEY FRAME (Image 2) -- a list of concrete,
visually verifiable conditions that hold in Image 2.

HOW TO WRITE GOOD CRITERIA:
- Keep every criterion as short and simple as possible.
- Every criterion's SUBJECT is a TOWEL named by its COLOR. Do NOT write criteria about the
rod, gripper, or table as the subject.
- When multiple towels share the same place, add each towel's RELATIVE position among those
towels.
- In the key frame each towel is in exactly ONE state:
    * on the table,
    * hanging over the rod,
    * grasped by the gripper,
    * not in the view.
- ONLY write a criterion you are VISUALLY CERTAIN about. If a towel is too occluded or its
state is ambiguous, OMIT it rather than guess.
- If a towel from Image 1 is clearly absent in the key frame, state that it is not in the
view.
- "Hanging over the rod" means supported by or draped over the rod, not lying flat on the
table.
- Each criterion is ONE short, self-contained sentence.
- Do not pad or guess.

GOOD EXAMPLES:
- "The red towel is on the table on the left."
- "The blue towel is on the table on the right."
- "The green towel is hanging over the rod on the left."
- "The yellow towel is hanging over the rod on the right."
- "The blue towel is grasped by the gripper."
- "The red towel is not in the view."

OUTPUT CONTRACT: Respond with STRICTLY a JSON array of strings and NOTHING else.
Example: ["...", "..."]

[AUTHORING USER PROMPT]

Image 1 is the episode's first frame (the reference for the COLOR identity of every towel).
Image 2 is the key frame to annotate.

Using Image 1 to fix the identity of each towel, write the success_criteria that hold in the
key frame (Image 2). Respond with ONLY a JSON array of strings.


--------------------------------------------------------------------------------

[JUDGING SYSTEM PROMPT]
```

```
You are an expert physical-status judge for a robot TOWEL hang / unhang benchmark. A robot
picks a towel up from the table and hangs / drapes it over a horizontal hanging rod, or
takes a towel off the rod and puts it down on the table.

You are shown one key frame plus success criteria. Each criterion names a towel by color
and, when necessary, position, and asserts that it is on the table, hanging over the rod,
grasped near the table, grasped near the rod, or not visible.

For each criterion, return 1 if it is clearly satisfied and 0 if it is clearly not satisfied.

IDENTITY:
- Use the first frame and complete towel inventory to resolve identity.
- Match primarily by COLOR and then by relative position.
- Do not demand a fixed towel shape, because folds, draping, occlusion, and viewpoint change
its appearance.
- When two towels share a color, use relative position to distinguish them.

WHAT TO VERIFY:
- Whether the towel is on the table, hanging over the rod, grasped near the table, grasped
near the rod, or absent.
- Interpret left/middle/right relative to other towels in the same place.
- "Hanging over the rod" means visibly supported by or draped over the rod.
- For a grasp criterion, verify both the grasp and, if specified, proximity to the table or
rod.
- For "not visible", verify that no matching towel is visible.

Reserve 0 for clear contradictions: wrong towel, wrong location, present when it should be
absent, or absent when it should be present. Shape or fold differences alone do not
constitute failure.

Return exactly:
{
  "aspect_scores": [0 or 1, ...],
  "reasoning": ["brief visible evidence", "..."],
  "explanation": "one sentence summary"
}

Return exactly one score per criterion, in the same order.
```

Pour Water: Physical Criteria Authoring and Judging

```
[AUTHORING SYSTEM PROMPT]

You are an expert annotator for a robot manipulation benchmark.

SETTING: A robot arm grasps a cup that holds colored water and pours the water into another
cup. The scene's objects are CUPS, and the WATER is colored.

YOU ARE GIVEN TWO IMAGES:
- Image 1 is the episode's FIRST frame. Use it to establish every cup's COLOR and the COLOR
of the water.
- Image 2 is the KEY FRAME you must annotate.

IDENTITY VS POSITION:
- A cup's stable identity is its COLOR. Take each cup's color and water color from Image 1.
- A cup's POSITION is dynamic. Read its current position from Image 2 rather than copying
its initial position.
- Use Image 1 for identity and Image 2 for current position and state.
- If cup color cannot be reliably determined, name the cup by its relative position instead
of guessing a color.

YOUR JOB: Write the 'success_criteria' for the KEY FRAME.

HOW TO WRITE GOOD CRITERIA:
- First ground the key frame by comparing all cups and determining their relative
arrangement.
- Write one criterion per cup.
- Each criterion describes:
    (a) the cup's pose / position, and
    (b) the water inside the cup.
- Positions are relative to other cups, using terms such as left, middle, right, front,
behind, front-left, or back-right.
- Name each cup by color plus relative position when color is reliable.
- If color is unreliable, use relative position alone.
```

```
- Cup pose must be one of:
    * standing upright on the table,
    * grasped / held by the gripper,
    * tilted and pouring into another cup.
- Water content must be one of:
    * is empty,
    * contains <color> water.
- Combine cup pose and water content into one sentence.
- Only write states that are visually certain.
- Water content is binary. If colored liquid is clearly and obviously visible, write
"contains <color> water"; otherwise treat the cup as empty.
- When pouring, identify both source and target cups.
- Add a spill criterion only for a CLEAR colored spill or puddle on the table.
- Do not annotate minor droplets, wet sheen, or ambiguous regions as spills.

GOOD EXAMPLES:
- "The dark gray cup in the middle is standing upright on the table and is empty."
- "The clear cup on the right is standing upright on the table and contains red water."
- "The blue cup on the left is grasped by the gripper and contains red water."
- "The blue cup on the left is tilted over the white cup on the right, pouring red water
into it."
- "Red water is spilled on the table."

OUTPUT CONTRACT: Respond with STRICTLY a JSON array of strings and NOTHING else.
Example: ["...", "..."]

[AUTHORING USER PROMPT]

Image 1 is the episode's first frame and is the reference for each cup's COLOR identity, not
its position. Image 2 is the key frame to annotate.

Use Image 1 only to fix cup color identity. Read each cup's CURRENT position and state from
Image 2. If cup color is unclear, name it by position only. Respond with ONLY a JSON array
of strings.


--------------------------------------------------------------------------------

[JUDGING SYSTEM PROMPT]

You are an expert physical-status judge for a robot water-pouring benchmark.

Each criterion usually describes both a cup's pose or position and the water inside that
cup. For each criterion, return 1 only if every asserted component holds; otherwise return 0.

IDENTITY AND POSITION:
- Use Image 1 only to recall cup and water colors.
- Cup positions are dynamic and relative. Judge them from Image 2 by comparing cups to one
another.
- Identify a cup primarily by relative position and pouring role. Cup color is only a weak
secondary cue.
- Do not score 0 solely because cup color appears different under lighting, transparency, or
colored liquid.
- Judge water state, cup pose, and pouring relation as the primary signals.

WHAT TO VERIFY:
- GRASP: which cup the gripper visibly contacts and holds.
- POUR RELATION: whether the source cup is tilted above and over the rim of the specified
target.
- CUP POSE: upright on the table, grasped, or tilted.
- WATER CONTENT: empty or containing the stated colored water.
- SPILL: a clear colored spill or puddle on the table.

EMPTY VS CONTAINS WATER:
- If the criterion says the cup is empty, default to empty unless water is especially
obvious.
- If the criterion says the cup contains water, accept it unless the cup is clearly empty.
- When liquid is clearly visible, its color must match the inventory.
- Do not infer water from droplets, residue, reflections, or ambiguous wet sheen.

For pouring, use the strongest visible cue: liquid in the target, a correct tilt-and-pour
pose, a visible stream or droplets, or a clearly changed water level.

Judge cup appearance leniently, but judge the asserted state and spatial relation strictly.

Return exactly:
{
  "aspect_scores": [0 or 1, ...],
```

```
  "reasoning": ["brief visible evidence", "..."],
  "explanation": "one sentence summary"
}

Return exactly one score per criterion, in the same order.
```

Pull Drawer: Physical Criteria Authoring and Judging

```
[AUTHORING SYSTEM PROMPT]

You are an expert annotator for a robot manipulation benchmark.

SETTING: A robot with two grippers OPENS or CLOSES a drawer. The drawers form a single
vertical column of three drawers, and each drawer is a different color. A drawer is either
OPEN or CLOSED, and an open drawer may contain an object.

YOU ARE GIVEN TWO IMAGES:
- Image 1 is the episode's FIRST frame. Use it to establish the color of each drawer and any
object inside an open drawer.
- Image 2 is the KEY FRAME you must annotate.

YOUR JOB: Write the 'success_criteria' for Image 2.

HOW TO WRITE GOOD CRITERIA:
- Identify each drawer by COLOR ONLY.
- Do not use top, middle, bottom, left, or right in criteria.
- If a drawer is open and empty, write: "The <color> drawer is open."
- If a drawer is open and contains an object, write two separate criteria:
    1. "The <color> drawer is open."
    2. "A <object> is inside the <color> drawer."
- Name an inside object by color plus rough kind.
- Do not over-specify its exact category.
- If more than one drawer is open, write one criterion per open drawer.
- If no drawer is open, write exactly: "All drawers are closed."
- Judge drawer open / closed state strictly.
- If the inside object is ambiguous, omit the object criterion rather than guessing.
- Each criterion must be one short, self-contained sentence.

GOOD EXAMPLES:
- "The dark green drawer is open."
- "A green fruit is inside the dark green drawer."
- "The red drawer is open."
- "All drawers are closed."

OUTPUT CONTRACT: Respond with STRICTLY a JSON array of strings and NOTHING else.
Example: ["...", "..."]

[AUTHORING USER PROMPT]

Image 1 is the episode's first frame and provides the color of each drawer and any object
placed inside a drawer. Image 2 is the key frame to annotate.

Using Image 1 to fix the colors, write which colored drawer is open, what object is inside
it as a separate criterion, or that all drawers are closed. Use drawer COLOR only and no
position words. Respond with ONLY a JSON array of strings.


--------------------------------------------------------------------------------

[JUDGING SYSTEM PROMPT]

You are an expert physical-status judge for a robot drawer open / close benchmark.

The scene contains a vertical column of three differently colored drawers. An open drawer
may contain an object.

For each criterion, return 1 when it is visibly satisfied and 0 when it is clearly not
satisfied.

IDENTITY:
- Use Image 1 as the reference frame.
- Use the object inventory to align drawer and inside-object names.
- For an object inside a drawer, judge identity leniently by color and rough shape. Do not
require an exact object category.
```

```
WHAT TO VERIFY:
- Judge drawer OPEN / CLOSED state strictly.
- Mark a drawer as open only with clear geometric evidence that the drawer body is pulled
out, such as:
    * a visible inner cavity or interior volume,
    * exposed side rails or a clear gap,
    * clear forward protrusion from the cabinet frame.
- A visible handle, shadow, slight perspective difference, or gripper contact is not
sufficient evidence of opening.
- If strong open cues are absent, treat the drawer as closed.
- For "All drawers are closed", return 1 when no drawer has strong open evidence.
- Evaluate drawer-state and object-inside criteria independently.
- For an inside-object criterion, require the same drawer to be visibly open and the object
inside to match by color and rough shape.
- If the drawer is closed or its opening state is unclear, score the inside-object criterion
as 0.

Reserve 0 for a wrong drawer state, a wrong drawer being open, a drawer being closed when it
should be open, or a missing / clearly mismatched inside object.

Return exactly:
{
  "aspect_scores": [0 or 1, ...],
  "reasoning": ["brief visible evidence", "..."],
  "explanation": "one sentence summary"
}

Return exactly one score per criterion, in the same order.
```

Ground-truth consistency check. We verify the internal consistency of the complete authoring–judging pipeline by passing each real-world outcome keyframe used for criteria authoring to the physical-state judge. GPT-5.5 judges 100% of the authored criteria as satisfied on these ground-truth keyframes. This consistency check provides high confidence that the VLM evaluator can reliably recognize the annotated physical states and apply the criteria consistently when computing physical state correctness.

F.2 Interaction Region Annotation

Physical-state criteria are attached to outcome keyframes, while interaction regions are annotated frame-wise throughout each evaluation episode. We use SAM3 (Carion et al., 2026) to generate temporal mask sequences for the manipulated objects and the robot arms, end effectors, or human hands. For camera view v and source $s \in \{\mathrm{obj}, \mathrm{arm}\}$, the annotation pipeline produces $L_i^{v,s}$ temporal mask tracks:

$$\mathcal{M}_i^{v,s} = \left\{ \left(M_{i,t}^{v,s,m}\right)_{t=1}^{T_i} \right\}_{m=1}^{L_i^{v,s}}, \tag{S1}$$

where

$$M_{i,t}^{v,s,m} \in \{0,1\}^{H \times W} \tag{S2}$$

is the binary mask of track m at frame t. The index m identifies an individual entity track within source s. For example, different fruits or towels are represented by different object-mask tracks, while the left and right robot arms are represented by different arm-mask tracks. Object masks cover task-relevant entities whose states change under the actions, including fruits, cups, liquid regions, towels, drawer components, and other manipulated objects. Arm masks cover robot arms and grippers or, for the human-hand embodiment, the interacting hands. The masks are produced for every camera view. The masks for the main camera view are used for interaction spatial alignment and key interaction area fidelity, whereas the masks for all camera views are used for multi-view action coherence, evaluating models that support multi-view outputs.

G Additional Details on WorldOracle Metrics

This section provides the complete definitions and implementation details of the four evaluation dimensions introduced in Sec. 3.2.

G.1 PHYSICAL STATE CORRECTNESS

We use GPT-5.5 (OpenAI, 2026) as the VLM evaluator to assess the physical correctness for predicted outcome keyframes of each interaction segment. This metric is computed using the primary `top_head` view.

Physical state criteria. For interaction segment n in episode i, we define a set of observable physical-state criteria:

$$\mathcal{C}_{i,n} = \{c_{i,n,j}\}_{j=1}^{J_{i,n}}, \tag{S3}$$

where $J_{i,n}$ is the number of criteria associated with the segment's outcome keyframe. These criteria describe task-specific physical states, including object identity and location, grasp state, articulation state, containment, contact, and liquid transfer. Examples include "the yellow fruit is held by the gripper" and "the blue towel is hanging over the rack". As described in Sec. F.1, these criteria are authored from the corresponding real-world outcome keyframe and verified through our ground-truth consistency check.

VLM evaluation. Let $v_0 = \texttt{top_head}$ denote the primary camera view. For each interaction segment, the VLM receives:

- the predicted outcome keyframe $\hat{K}_{i,n}^{v_0}$;
- the initial real observation $O_{i,0}^{v_0}$ as an object-identity reference;
- the episode-level object inventory; and
- the physical-state criteria $\mathcal{C}_{i,n}$.

We use task-specific judging prompts for *pick fruit*, *hang towel*, *pour water*, and *pull drawer*. The complete prompts are provided in Sec. F.1. For each criterion $c_{i,n,j}$, the VLM returns a binary judgment $y_{i,n,j} \in \{0, 1\}$, where $y_{i,n,j} = 1$ indicates that the criterion is visibly satisfied in the predicted outcome keyframe.

Score aggregation. The physical state correctness score of an outcome keyframe is the mean of its criterion-level judgments:

$$S_{\text{phys},i,n} = \frac{1}{J_{i,n}} \sum_{j=1}^{J_{i,n}} y_{i,n,j}. \tag{S4}$$

The episode-level score, expressed on the common $[0, 100]$ scale, is then computed by averaging over its N_i outcome keyframes:

$$S_{\text{phys},i} = \frac{100}{N_i} \sum_{n=1}^{N_i} S_{\text{phys},i,n}. \tag{S5}$$

Then we average $S_{\text{phys},i}$ over all the episodes to obtain the final score S_{phys}. This formulation assigns equal weight to each interaction segment, irrespective of the number of criteria associated with its outcome keyframe. It also preserves partial correctness when only a subset of the physical-state criteria is satisfied.

G.2 INTERACTION SPATIAL ALIGNMENT

Interaction spatial alignment measures the spatial correspondence of manipulated objects and robot arms or human hands between real and predicted observations and calculates their spatial displacement. The metric evaluates the primary camera view $v_0 = \texttt{top_head}$ at a temporal stride of 10 frames. For compactness, we omit the view superscript v_0 for masks from this view. For example:

$$M_{i,t}^{s,m} := M_{i,t}^{v_0,s,m}. \tag{S6}$$

Dense feature extraction. For sampled frame t of episode i, we first extract dense DINOv3 (Siméoni et al., 2025) feature maps from the real and predicted observations in the primary view:

$$F_{i,t} = \phi_{\text{DINO}}\left(O_{i,t}^{v_0}\right), \qquad \hat{F}_{i,t} = \phi_{\text{DINO}}\left(\hat{O}_{i,t}^{v_0}\right), \tag{S7}$$

where ϕ_{DINO} is DINOv3 ViT-B/16 in our implementation, which produces spatial feature maps:

$$F_{i,t}, \hat{F}_{i,t} \in \mathbb{R}^{D_f \times H_p \times W_p}, \tag{S8}$$

where D_f is the feature dimension and $H_p \times W_p$ is the spatial patch grid. Each patch feature is L_2-normalized before matching. For source $s \in \{\mathrm{obj}, \mathrm{arm}\}$, let $M_{i,t}^{s,m} \in \{0,1\}^{H \times W}$ denote the m-th ground-truth mask at frame t. Here, m indexes individual mask tracks within the object or robot-arm source. Because $M_{i,t}^{s,m}$ is defined in pixel space, we resize it to the DINOv3 patch grid using nearest-neighbor interpolation:

$$\widetilde{M}_{i,t}^{s,m} = \mathrm{Resize}_{\mathrm{NN}}\left(M_{i,t}^{s,m}, H_p, W_p\right). \tag{S9}$$

The resized mask identifies a set of patch locations in the real observation:

$$\mathcal{Q}_{i,t}^{s,m} = \left\{ u \;\middle|\; \widetilde{M}_{i,t}^{s,m}(u) = 1 \right\}. \tag{S10}$$

We use the corresponding real-observation features as template tokens:

$$\mathcal{Z}_{i,t}^{s,m} = \left\{ F_{i,t}(u) \;\middle|\; u \in \mathcal{Q}_{i,t}^{s,m} \right\}. \tag{S11}$$

Each template token is then used to search over the entire feature map of the corresponding predicted observation. Masks containing fewer than two patch tokens are considered too small for reliable matching and are excluded.

Feature correspondence.

For each template location $u \in \mathcal{Q}_{i,t}^{s,m}$, we find its nearest neighbor over all locations u' in the complete predicted feature map:

$$\hat{u} = \arg\max_{u'} \mathrm{sim}\left(F_{i,t}(u), \hat{F}_{i,t}(u')\right), \tag{S12}$$

where cosine similarity is defined as

$$\mathrm{sim}(z, \hat{z}) = \frac{z^\top \hat{z}}{\|z\|_2 \|\hat{z}\|_2}. \tag{S13}$$

A correspondence $(u, \hat{u})$ is mutual if u is also the nearest template token to $\hat{u}$ in the reverse direction. We retain mutual nearest-neighbor correspondences when at least three mutual matches are available. When fewer than three mutual matches are found, we use all forward nearest-neighbor matches. The subsequent confidence score still depends on the number and ratio of mutual matches, and therefore penalizes correspondences with weak mutual support.

Spatial displacement. Let $x(u) \in \mathbb{R}^2$ denote the center of patch u in the original image coordinate system. For each retained correspondence $(u_\nu, \hat{u}_\nu)$, we compute its displacement vector

$$\boldsymbol{\delta}_\nu = x(\hat{u}_\nu) - x(u_\nu) \tag{S14}$$

and displacement magnitude

$$d_\nu = \|\boldsymbol{\delta}_\nu\|_2. \tag{S15}$$

The spatial displacement for mask $M_{i,t}^{s,m}$ is the median displacement across retained correspondences:

$$d_{i,t}^{s,m} = \mathrm{median}_\nu \, d_\nu. \tag{S16}$$

Using the median reduces sensitivity to isolated incorrect feature matches.

Feature confidence. A small estimated displacement is meaningful only if the underlying feature correspondences are reliable. We characterize correspondence reliability using four signals.

First, the mean cosine similarity over the retained matches is

$$\bar{c} = \frac{1}{R} \sum_{\nu=1}^{R} \mathrm{sim}\left(F_{i,t}(u_\nu), \hat{F}_{i,t}(\hat{u}_\nu)\right), \tag{S17}$$

where R is the number of retained matches.

Second, let $c_\nu^{(1)}$ and $c_\nu^{(2)}$ denote the highest and second-highest similarities obtained for template location u_ν. The matching margin is the mean difference between the two predicted similarities:

$$\Delta = \frac{1}{R}\sum_{\nu=1}^{R}\left(c_\nu^{(1)} - c_\nu^{(2)}\right). \tag{S18}$$

A larger margin indicates that the best correspondence is more clearly distinguished from alternative matches.

Third, the mutual-match ratio is

$$r_{\mathrm{mut}} = \frac{R_{\mathrm{mut}}}{\left|\mathcal{Q}_{i,t}^{s,m}\right|}, \tag{S19}$$

where R_{mut} is the number of mutual matches among all template tokens.

Finally, we measure whether the matched features agree on a common geometric translation. We estimate the median translation

$$\boldsymbol{\delta}_{\mathrm{med}} = \mathrm{median}_\nu\, \boldsymbol{\delta}_\nu = (\mathrm{median}_\nu\, \delta_{\nu,x}, \mathrm{median}_\nu\, \delta_{\nu,y}) \tag{S20}$$

and define the geometric inlier ratio as

$$r_{\mathrm{inl}} = \frac{1}{R}\sum_{\nu=1}^{R}\mathbf{1}\left[\|\boldsymbol{\delta}_\nu - \boldsymbol{\delta}_{\mathrm{med}}\|_2 \le r_{\mathrm{geo}}\right]. \tag{S21}$$

We use an inlier radius of $r_{\mathrm{geo}} = 32$ pixels. We combine these 4 signals into a feature-confidence score:

$$c_{\mathrm{feat}} = \sigma\left(b + w_{\cos}\bar{c} + w_\Delta\Delta + w_{\mathrm{mut}}r_{\mathrm{mut}} + w_{\mathrm{inl}}r_{\mathrm{inl}}\right), \tag{S22}$$

where $\sigma(\cdot)$ is the sigmoid function. We use

$$b = -2, \quad w_{\cos} = 3, \quad w_\Delta = 8, \quad w_{\mathrm{mut}} = 1.5, \quad w_{\mathrm{inl}} = 1.5. \tag{S23}$$

Match support. A high feature-confidence score may still be unreliable if it is supported by only a small number of mutual matches. We therefore define a match-support score:

$$c_{\mathrm{sup}} = \sqrt{\min\left(\frac{r_{\mathrm{mut}}}{\rho}, 1\right)\left(1 - \exp\left[-\frac{R_{\mathrm{mut}}}{\kappa}\right]\right)}. \tag{S24}$$

The first term measures mutual-match coverage, while the second measures the absolute number of mutual matches. We use

$$\rho = 0.3, \qquad \kappa = 8. \tag{S25}$$

The final correspondence confidence is

$$c = c_{\mathrm{feat}}c_{\mathrm{sup}}^{\eta}, \tag{S26}$$

where $\eta = 1$ in our implementation.

Alignment score. We map correspondence confidence to a bounded quality score:

$$q_c = \mathrm{clip}\left(\frac{c - c_{\mathrm{bad}}}{c_{\mathrm{good}} - c_{\mathrm{bad}}}, 0, 1\right). \tag{S27}$$

Similarly, we map the median pixel displacement to a bounded distance score:

$$q_d = \mathrm{clip}\left(\frac{d_{\mathrm{bad}} - d_{i,t}^{s,m}}{d_{\mathrm{bad}} - d_{\mathrm{good}}}, 0, 1\right). \tag{S28}$$

The mask-level ISA score is

$$A_{i,t}^{s,m} = 100 q_d q_c^{\gamma}. \tag{S29}$$

The confidence exponent $\gamma > 1$ prevents an ambiguous feature match from receiving a high alignment score merely because its estimated displacement is small. We use

$$c_{\mathrm{bad}} = 0.5, \qquad c_{\mathrm{good}} = 0.8, \tag{S30}$$

$$d_{\text{good}} = 30 \text{ pixels}, \qquad d_{\text{bad}} = 150 \text{ pixels}, \qquad \gamma = 3. \tag{S31}$$

Score aggregation. Let $\mathcal{V}_{i,t}^s$ denote the valid masks for source s at frame t. We first average the mask-level scores within each source:

$$A_{i,t}^s = \frac{1}{|\mathcal{V}_{i,t}^s|} \sum_{m \in \mathcal{V}_{i,t}^s} A_{i,t}^{s,m}. \tag{S32}$$

Let $\mathcal{T}_i^s$ denote the sampled frames containing at least one valid mask for source s. The episode-level ISA score is

$$A_i = \frac{\sum_s \sum_{t \in \mathcal{T}_i^s} A_{i,t}^s}{\sum_s |\mathcal{T}_i^s|}. \tag{S33}$$

Thus, every valid source–frame pair contributes equally. When both sources are valid at every sampled frame, this is equivalent to averaging the object and robot-arm scores per frame and then averaging over time. The benchmark-level ISA score is the mean over evaluated episodes:

$$A = \frac{1}{N} \sum_{i=1}^{N} A_i. \tag{S34}$$

We evaluate the top-head view at a temporal stride of 10 frames.

G.3 Key Interaction Area Fidelity

Let $v_0 = \texttt{top_head}$ denote the primary view. For compactness, we omit the view superscript for masks from this view:

$$M_{i,t}^{s,m} := M_{i,t}^{v_0,s,m}. \tag{S35}$$

Whole-frame visual metrics can be dominated by static background regions and may therefore overlook visual errors around manipulated objects, robot arms, and contact regions. We thus evaluate visual quality specifically within these action-relevant regions.

Interaction-region extraction. For frame t of episode i, source $s \in \{\text{obj}, \text{arm}\}$, and mask instance m, let $M_{i,t}^{s,m} \in \{0,1\}^{H \times W}$ denote the corresponding ground-truth segmentation mask. Object masks identify manipulated objects, while robot-arm masks identify the robot body, end effector, or human hand. To include object boundaries and nearby contact information, we dilate each mask by δ pixels:

$$\bar{M}_{i,t}^{s,m} = \text{Dilate}\left(M_{i,t}^{s,m}; \delta\right). \tag{S36}$$

We use $\delta = 5$ pixels in our experiments. Masks containing fewer than 10 foreground pixels before dilation are excluded. We then extract the bounding box of the dilated mask, expand it by a padding of 10 pixels, and ensure that both its height and width are at least 64 pixels. Let $B_{i,t}^{s,m}$ denote the resulting crop operator. The cropped and background-masked ground-truth and predicted observations are

$$X_{i,t}^{s,m} = B_{i,t}^{s,m}\left(O_{i,t} \odot \bar{M}_{i,t}^{s,m}\right), \quad \hat{X}_{i,t}^{s,m} = B_{i,t}^{s,m}\left(\hat{O}_{i,t} \odot \bar{M}_{i,t}^{s,m}\right). \tag{S37}$$

Pixels inside the dilated interaction region retain their RGB values, while pixels outside the region are set to zero. Cropping prevents the shared black background from dominating the perceptual measurements.

Masked PSNR. We compute PSNR (Huynh-Thu & Ghanbari, 2008) directly over pixels inside the dilated mask:

$$p_{i,t}^{s,m} = 10 \log_{10}\left(\frac{1}{\text{MSE}_M\left(\hat{O}_{i,t}, O_{i,t}; \bar{M}_{i,t}^{s,m}\right)}\right). \tag{S38}$$

For observations normalized to $[0,1]$, the masked mean squared error is

$$\text{MSE}_M\left(\hat{O}, O; M\right) = \frac{\sum_{x,y} M(x,y) \left\|\hat{O}(x,y) - O(x,y)\right\|_2^2}{3 \sum_{x,y} M(x,y)}. \tag{S39}$$

This calculation excludes all pixels outside the interaction region.

Masked LPIPS. We compute LPIPS (Zhang et al., 2018) between the cropped and background-masked observations:

$$l_{i,t}^{s,m} = \mathrm{LPIPS}\left(\hat{X}_{i,t}^{s,m}, X_{i,t}^{s,m}\right). \tag{S40}$$

Masked V-JEPA similarity. We additionally use V-JEPA 2 features (Assran et al., 2025) to measure representation-level similarity between the same masked crops:

$$v_{i,t}^{s,m} = \cos\left(\phi_{\mathrm{VJEPA}}\left(\hat{X}_{i,t}^{s,m}\right), \phi_{\mathrm{VJEPA}}\left(X_{i,t}^{s,m}\right)\right). \tag{S41}$$

We use the V-JEPA 2 ViT-B encoder and compute cosine similarity between the pooled representations of the real and predicted masked crops.

Metric normalization. The three measurements have different ranges and directions. For a higher-is-better metric, we define

$$\mathcal{N}_+(x; a, b) = \mathrm{clip}\left(\frac{x-a}{b-a}, 0, 1\right). \tag{S42}$$

For a lower-is-better metric, we define

$$\mathcal{N}_-(x; a, b) = \mathrm{clip}\left(\frac{b-x}{b-a}, 0, 1\right). \tag{S43}$$

We use the empirical normalization ranges $[a_p, b_p] = [8, 30]$ for PSNR, $[a_l, b_l] = [0.1, 0.35]$ for LPIPS, and $[a_v, b_v] = [0.97, 1.0]$ for V-JEPA similarity.

Score aggregation. Each metric is normalized before any averaging. The quality score for mask m is the geometric mean of the three normalized measurements:

$$q_{i,t}^{s,m} = 100\left[\mathcal{N}_+\left(p_{i,t}^{s,m}; a_p, b_p\right)\mathcal{N}_-\left(l_{i,t}^{s,m}; a_l, b_l\right)\mathcal{N}_+\left(v_{i,t}^{s,m}; a_v, b_v\right)\right]^{\frac{1}{3}}. \tag{S44}$$

The geometric mean requires the prediction to perform well across pixel-level fidelity, perceptual similarity, and representation-level similarity. A low score in any one component therefore reduces the combined mask-level score.

Let $\mathcal{M}_{i,t}^{s}$ denote the valid masks belonging to source s at frame t. We first average the mask-level scores within each source:

$$q_{i,t}^{s} = \frac{1}{|\mathcal{M}_{i,t}^{s}|}\sum_{m\in\mathcal{M}_{i,t}^{s}} q_{i,t}^{s,m}. \tag{S45}$$

The object and robot-arm scores are then averaged with equal weight at each frame:

$$q_{i,t} = \frac{1}{2}\left(q_{i,t}^{\mathrm{obj}} + q_{i,t}^{\mathrm{arm}}\right). \tag{S46}$$

Finally, the episode-level key interaction area fidelity score is the mean over its sampled frames:

$$Q_i = \frac{1}{|\mathcal{T}_i|}\sum_{t\in\mathcal{T}_i} q_{i,t}. \tag{S47}$$

This frame-first aggregation gives equal weight to the object and robot-arm regions at every time step. We evaluate the top-head view at a temporal stride of 10 frames.

G.4 Multi-View Action Coherence

For models that support multi-view output, a predicted rollout may remain aligned with reality in one view while producing an incorrect robot pose or motion in another. We therefore evaluate robot-execution alignment across three synchronized camera views: $\mathcal{V} = \{\texttt{top_head}, \texttt{hand_left}, \texttt{hand_right}\}$.

Per-view robot-arm alignment. We evaluate only the robot-arm source because manipulated objects are frequently occluded, absent, or only partially visible in the auxiliary hand-camera views. Robot-arm motion provides a more consistently observable reference across the three views.

Table S3: **Training configurations for the evaluated ACWMs and policy.** *Pretrained embodiment(s)* denotes the robot or embodiment domains covered by the official checkpoint used for initialization. Batch denotes the effective global batch size across eight NVIDIA H100 GPUs. The reported step corresponds to the checkpoint used for evaluation.

Model	Pretrained Embodiment(s)	Step	Batch	Optimizer	Peak LR	Weight Decay	Precision
IRASim	WidowX (Bridge)	100K	32	AdamW	1×10^{-4}	0	FP32
Ctrl-World	Franka Panda	250K	32	AdamW	1×10^{-5}	1×10^{-2}	BF16
DreamDojo	Human, Unitree G1, Fourier GR-1, AgiBot, YAM	100K	16	Fused AdamW	5×10^{-6}	1×10^{-1}	BF16
Cosmos 3	AgiBot, Franka Panda, Google Robot, WidowX, UMI, UR	200K	8	AdamW	2×10^{-5}	5×10^{-2}	BF16
GE-Sim 2.0	AgiBot G1	23K	64	8-bit AdamW	1×10^{-4}	5×10^{-5}	BF16
$\pi_{0.5}$ policy	Franka Panda	51K	256	AdamW	2.5×10^{-5}	1×10^{-10}	BF16

Table S4: **Architectural and interface configurations.** *Generated Frames* denotes the number of future frames produced per model invocation, while *History Frames* denotes the number of previous RGB frames used as visual context. The *Output* column indicates whether the model natively generates synchronized multi-view observations.

Model	Size	Action Conditioning	Generated Frames	History Frames	Output	Resolution
IRASim	0.7B	Relative EEF/gripper; frame-level AdaLN	15	1	Video	320×256
Ctrl-World	1.5B	Absolute EEF/gripper; frame-wise cross-attention	5	6 (sparse)	Multi-view video	384×256
DreamDojo	2B	Block-relative robot actions; 4-action chunks via AdaLN	12	1	Video	640×480
Cosmos 3	16B	Relative EEF/gripper/hand pose; action-token conditioning	16	1	Video	832×480
GE-Sim 2.0	2B	Absolute EEF/gripper; channel-concat trajectory + ray maps	25	4 (sparse)	Multi-view video, proprioception, reward	512×384
JointControlVideo	14B	Absolute hand pose; channel-concat depth-aware heatmaps	80	1	Video	832×480
OSCAR	2B	Absolute hand pose; channel-concat rendered skeleton	80	1	Video	640×480

For view $v \in \mathcal{V}$, mask track m, and sampled frame t, we compute the mask-level alignment score $A_{i,t}^{v,\mathrm{arm},m}$ using the procedure described in Sec. G.2. Let

$$\mathcal{I}_{i,t}^{v} = \left\{ m \;\middle|\; A_{i,t}^{v,\mathrm{arm},m} \text{ is valid} \right\} \tag{S48}$$

denote the valid robot-arm mask indices in view v at frame t. The available per-view alignment score is

$$A_{i,t}^{v} = \frac{1}{|\mathcal{I}_{i,t}^{v}|} \sum_{m \in \mathcal{I}_{i,t}^{v}} A_{i,t}^{v,\mathrm{arm},m}, \qquad |\mathcal{I}_{i,t}^{v}| > 0. \tag{S49}$$

Cross-view aggregation. The frame-level multi-view action coherence score is the average of the three view-level robot-arm alignment scores:

$$M_{i,t} = \frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} \widetilde{A}_{i,t}^{v}. \tag{S50}$$

A missing or poorly aligned view directly reduces the score because every view receives equal weight. Let $\mathcal{T}_i$ denote the frame indices. The episode-level score is

$$S_{\mathrm{mv},i} = \frac{1}{|\mathcal{T}_i|} \sum_{t \in \mathcal{T}_i} M_{i,t}. \tag{S51}$$

H Training and Implementation Details

We conduct training experiments on 8 NVIDIA H100 GPUs, with training configurations shown in Table S3 and architectural configurations shown in Table S4.

IRASim. We initialize IRASim-XL from the released Bridge frame-Ada checkpoint. We extend the original 7D relative single-arm end-effector representation to a 14D dual-arm action interface.

Each action is embedded independently and injected into the spatial transformer blocks. Training uses a per-GPU batch size of 2 with two-step gradient accumulation, resulting in a global batch size of 32 across 8 GPUs. We use AdamW with a learning rate of 1×10^{-4}, zero weight decay, gradient clipping at 0.1, and an EMA decay of 0.9999. The model generates 15 future frames per autoregressive chunk at a resolution of 320×256, conditioned on 1 clean history frame.

Ctrl-World. We extend the original 7D absolute single-arm end-effector representation to a 14D dual-arm action interface. Each action is projected by a three-layer MLP and supplied as the cross-attention conditioning of the SVD spatial transformer. Training uses a per-GPU batch size of 4 without gradient accumulation, giving a global batch size of 32 across 8 GPUs. We use AdamW with a learning rate of 1×10^{-5} and bfloat16 precision. The model generates 5 future frames per autoregressive chunk at a resolution of 384×256, conditioned on 6 sparsely sampled history frames.

DreamDojo. We initialize DreamDojo from its 2B AgiBot post-training checkpoint. We follow the official embodiment-specific action-conditioning interface. Robot actions are converted into block-relative representations, grouped into four-action chunks to match the temporal compression of the video tokenizer, and projected into the adaptive normalization layers of the DiT backbone. Training uses a per-GPU batch size of 2, resulting in a global batch size of 16 across 8 GPUs. We use fused AdamW with a learning rate of 5×10^{-6}, weight decay of 0.1, and bfloat16 precision. The model generates 12 future frames per autoregressive chunk at a resolution of 640×480, conditioned on one preceding frame.

Cosmos 3. We initialize Cosmos 3 Nano from the official 16B checkpoint. We replace the pretrained 29D AgiBot action interface with a 20D dual-arm representation comprising, for each arm, a 3D relative translation, a 6D relative rotation, and a scalar gripper state. The resulting actions are quantile-normalized and padded to the model's unified 64D action space. Training uses a per-GPU batch size of 1 without gradient accumulation, resulting in a global batch size of 8 across 8 GPUs. We use AdamW with a learning rate of 2×10^{-5} and bfloat16 precision. The model generates 16 future frames per autoregressive chunk at a resolution of 832×480, conditioned on one preceding frame.

GE-Sim 2.0. We initialize GE-Sim 2.0 from its 2B AgiBot post-training backbone, which jointly generates synchronized top-head, left-hand, and right-hand views. GE-Sim 2.0 conditions on actions geometrically. The absolute end-effector poses and gripper states of both arms are projected into every camera view to render pixel-aligned trajectory maps. The camera parameters are provided as Plücker ray maps. The ABC-130K training data contain only joint positions. We therefore recover the end-effector poses by forward kinematics on the public YAM URDF and calibrate the cameras of each episode from its videos. The wrist-camera mounts and focal lengths are fitted by aligning the robot's own fingertips in the image with their forward-kinematics positions. For the top camera, we reconstruct all views with VGGT-Ω (Wang et al., 2026) and align the reconstructed wrist-camera centers to the calibrated ones with a similarity transform. Training uses a per-GPU batch size of 8 without gradient accumulation, resulting in a global batch size of 64 across 8 GPUs. We use 8-bit AdamW with a constant learning rate of 1×10^{-4} after a 200-step linear warm-up, a weight decay of 5×10^{-5}, and bfloat16 precision. The model generates frames with a resolution of 512×384 per view and predicts 25 future frames per autoregressive chunk, conditioned on 4 sparsely sampled memory frames.

$\pi_{0.5}$ **Policy.** We initialize $\pi_{0.5}$ from its official DROID checkpoint and fine-tune it on our AgiBot training suite using 8 NVIDIA H100 GPUs. Training uses AdamW with a global batch size of 256 and bfloat16 precision. The learning rate is linearly warmed up to 2.5×10^{-5} over the first 1K steps and subsequently decayed to 2.5×10^{-6}. We use the checkpoint at 51K optimization steps to collect the policy-rollout distribution split.

I LIMITATIONS

Single real-world reference. Each trial uses one observed real-world execution as a reference. WorldOracle may penalize alternative outcomes that are physically plausible under stochastic contact, fluid, or deformable-object dynamics. Repeated executions from matched initial states would be required to estimate the corresponding real–real variability, which we leave to future work.

Dataset coverage. The current benchmark covers three embodiments and four manipulation tasks, but it does not represent the full diversity of robots, environments, and interaction types. Moreover, policy-rollout evaluation is currently limited to AgiBot G2. We plan to expand the YAM training set and collect policy-rollout trajectories for additional embodiments and tasks.

Unequal adaptation settings. The three embodiments have different levels of training support: AgiBot uses a dedicated training set, YAM relies on public data with a larger domain gap, and human-hand predictions are evaluated zero-shot. Their results therefore represent different levels of difficulty and should not be interpreted as a controlled comparison of cross-embodiment generalization. Future evaluations should use matched training data and adaptation budgets.

Observational model comparisons. The evaluated models differ in architecture, pretraining data, model scale, and action representation. Our results can reveal consistent performance patterns, but they cannot establish which design choice causes them. Controlled training and architectural ablations are required to verify these hypotheses.