

PAWBENCH: HOW FAR ARE WE FROM PROBABILISTICALLY ALIGNED WORLD MODELING?

Yuandong Pu^{1,2 *} Le Zhuo³ Sayak Paul⁴ Gabriel Jorge Menezes³
 Avram Đorđević³ Shiyang Li² Yifan Zhou¹ Bin Fu² Wenlong Zhang²
 Junjun He⁵ Qiao Yu² Yihao Liu^{2 †} Jinbo Xing⁶ Xi Chen^{7 †}
¹Shanghai Jiao Tong University ²Shanghai AI Laboratory ³Krea AI
⁴Huggingface ⁵Shanghai Innovation Institute ⁶Tongyi Lab ⁷The University of Hong Kong

ABSTRACT

Recent video generation models are increasingly framed as world models. Many physical processes can unfold in more than one valid way. Therefore, a world model should reproduce not only a plausible trajectory, but also the distribution of possible behaviors under the same initial observation and action. We call this distribution-level requirement **probabilistic alignment**. However, existing evaluations largely assess individual-video plausibility and do not test whether repeated generations recover the correct distribution. This raises a central question: **how far are current video generators from probabilistically aligned world modeling?** To answer it, we formalize probabilistic alignment as a distributional criterion for world models and introduce **PAWBench**, a benchmark for evaluating video generators as stochastic samplers of world dynamics. We further introduce **PAWEval**, an outcome-level protocol that converts repeated video rollouts into empirical distributions over possible physical behaviors. Across 50 scenarios and eleven current systems, no model consistently matches the reference probabilities while recovering the range of valid behaviors. Having established this gap, we test whether language prompts, initial noise sampling, or model training can reshape the model’s predictive distribution. We believe our work can serve as a foundation for future efforts to move towards probabilistically aligned world modeling.

Project page: <https://pawbench.github.io>

1 INTRODUCTION

Recent advances in video generation have substantially improved instruction following, visual quality, and temporal coherence (Google, 2025; Kling AI, 2026; ByteDance Seed, 2026b; Lightricks, 2026a; MiniMax, 2026). Given an observation and a prompt or action-like control signal, current systems can generate plausible continuations of a scene. Together, these advances (OpenAI, 2024; Google DeepMind, 2025a; Agarwal et al., 2026) have strengthened the case for viewing video generators not only as content creation tools, but also as visual world models.

Yet a world model must do more than render visually plausible rollouts. Because an observation reveals only part of the world state and an action does not specify all of its consequences, the generated rollouts must remain consistent with the scene’s underlying geometry and dynamics (Li & World Labs Team, 2026). When the underlying process is stochastic, the same initial observation and action can admit multiple physically valid futures rather than a single correct continuation (Babaeizadeh et al., 2018; Denton & Fergus, 2018). A model should therefore capture the conditional distribution over these futures, not merely render one plausible sample. We call this requirement **probabilistic alignment**: under a fixed initial observation and action, the induced distribution should reflect both which futures are physically possible and how likely each one is. Probabilistic alignment matters for interaction and planning (Ravi et al., 2025), because decisions depend on both the possible consequences of an action and their relative likelihoods.

*This work was done during his internship at Shanghai Artificial Intelligence Laboratory.

†Corresponding Authors

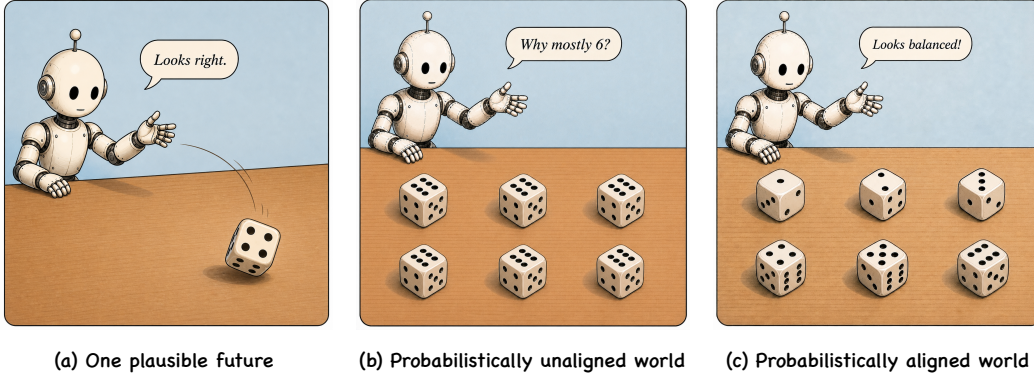


Figure 1: **One plausible future is not enough.** A single rollout from a video generation model can appear physically plausible, yet repeated rollouts from the same initial observation and action may reveal a probabilistically unaligned world: outcomes concentrate on a narrow subset of futures instead of matching the valid stochastic support. PAWBench evaluates this induced distribution or support over repeated futures, rather than judging a model by one sampled video.

Evaluation, however, still treats generated videos as isolated outputs rather than samples from the conditional distribution induced by the model. Existing benchmarks evaluate each generated video independently along dimensions of visual quality, temporal coherence, text or action alignment, and physical plausibility (Huang et al., 2024a;b; Bansal et al., 2024; Li et al., 2025). Yet success on these criteria does not reveal the model’s underlying distribution over possible futures given the same initial observation and action. A model may still produce plausible individual videos while collapsing to a narrow subset of valid outcomes or assigning probability mass in the wrong proportions. As illustrated in Fig. 1, one plausible future is therefore not enough. This leaves a central question: *how far are we from probabilistically aligned world modeling?*

To answer this question, we introduce **PAWBench** (Probabilistically Aligned World Bench), a diagnostic benchmark for evaluating whether video generators model the distribution of possible futures. PAWBench contains 50 scenarios spanning eight mechanism groups, organized into two complementary suites. *PAW-Calibration* covers 25 scenarios with analytically specified or symmetry-derived reference distributions, such as tossing a coin or spinning a wheel. *PAW-Coverage* covers 25 scenarios whose valid terminal outcomes can be enumerated but whose relative probabilities cannot be reliably specified, such as rolling a bowling ball or flipping a bottle. For each scenario, the source image and action prompt are held fixed across repeated rollouts. **PAWEval** maps readable, in-schema rollouts to terminal outcomes and records the remaining cases as outcome-readout failures. In PAW-Calibration, PAWBench compares the resulting empirical distribution with the reference probabilities; in PAW-Coverage, it measures recovery of the valid support.

We benchmark eleven current video generation models on PAWBench. No model achieves all three requirements: accurate outcome probabilities, broad coverage of valid futures, and reliable performance across scenes. Tests with more rollouts support the sampling budget used in the main evaluation, while PAWEval agrees with human judgments on videos with clear terminal outcomes. Controlled interventions show that model distributions change too little when the physical transition changes, yet can shift when only a non-causal cue changes. Current video generators therefore remain far from probabilistically aligned world modeling: individual videos may look plausible even when their underlying distributions miss valid futures and assign them the wrong probabilities.

We next move from measuring probabilistic alignment to asking whether it can be improved. We examine three points of intervention, moving from explicit language, through initial noise sampling, to the model’s learned predictive distribution under the same initial observation and action. Language makes possible futures explicit: VLMs predict what may happen, while prompts guide a video generator toward a named future. Sampling leaves the future unspecified and varies the initial noise to expose a broader range of possibilities from the same generator. Fine-tuning changes the model itself, testing whether it can learn how the probabilities of possible futures should vary with physical state.

Our main contributions are as follows.

- We formalize *probabilistic alignment* as a distributional criterion for world models and operationalize it through **PAWBench**, a 50-scenario diagnostic benchmark spanning eight physical mechanism groups under fixed initial observations and actions. **PAWEval** converts repeated video rollouts into empirical outcome distributions, enabling PAW-Calibration and PAW-Coverage to test probability-mass alignment and valid-support recovery, respectively.
- We benchmark current video generation models and reveal a failure hidden by current single-sample evaluation: plausible rollouts do not imply the correct distribution over possible futures. This gap persists beyond finite sampling and automated outcome readout.
- We probe probabilistic alignment from external guidance to model learning: we use language to predict and steer outcomes, sampling to broaden finite-budget exploration, and fine-tuning to reshape the learned distribution. These results distinguish inference-time steering and exploration from changes to the model’s learned predictive distribution.

2 PROBABILISTICALLY ALIGNED WORLD MODELING

World models represent distributions over possible futures. A world model should capture the range of futures that may unfold from a situation, not merely render one plausible continuation (Li & World Labs Team, 2026). Let x denote an initial observation and a an action. When the underlying process is stochastic, the same initial observation and action may lead to different physically valid outcomes. We therefore describe a video world model M through the conditional distribution $P_M(\tau \mid x, a)$ over the future trajectories τ it generates. Probabilistic alignment concerns both which futures this distribution supports and the relative probability it assigns to each one.

From a plausible future to an aligned distribution. Producing one plausible continuation meets only the weakest requirement: it shows that the model can realize one future. *Support alignment* asks whether it can realize the distinct outcomes available under the same condition, rather than collapsing to a subset. *Probability-mass alignment* is stronger: it asks whether those outcomes occur in the right proportions. A model can therefore be diverse without being aligned; covering every outcome does not guarantee assigning the correct probability to each.

To define these two levels of alignment, we map each generated trajectory to the terminal outcome it realizes. Let $\mathcal{Y}$ be the finite set of terminal outcomes, and let $g(\tau) \in \mathcal{Y}$ denote the outcome realized by trajectory τ . The model’s trajectory distribution then induces

$$p_M(y \mid x, a) = \Pr_{\tau \sim P_M(\cdot \mid x, a)}[g(\tau) = y].$$

This outcome distribution captures how much probability the model assigns to each possible result, while abstracting away differences among trajectories that reach the same result. When a reference distribution q over $\mathcal{Y}$ is available, probability-mass alignment requires $p_M(\cdot \mid x, a) = q$. When only the set of possible outcomes is known, support alignment requires $\text{supp}(p_M(\cdot \mid x, a)) = \mathcal{Y}$.

3 PAWBENCH

Having defined probabilistic alignment, we now describe how PAWBench makes this criterion measurable for video generators. The benchmark evaluates repeated rollouts under a fixed initial observation and action across two complementary alignment regimes, using an outcome-level evaluator. We first describe the overall benchmark design, including **PAW-Calibration** and **PAW-Coverage** (§3.1), then present the scenario construction process that makes repeated-rollout statistics meaningful (§3.2). We finally introduce **PAWEval**, a rubric-based outcome judging protocol that maps generated rollouts to terminal outcomes and converts them into distributional scores (§3.3).

3.1 BENCHMARK DESIGN

PAWBench evaluates a video generation model through repeated rollouts conditioned on the same initial observation and action. Each benchmark item provides a source image representing the initial



Figure 2: **PAWBench scenario taxonomy.** PAWBench covers eight mechanism groups under fixed initial observations and actions. PAW-Calibration contains calibrated probability scenarios with analytically specified reference distributions, while PAW-Coverage contains complex stochastic interactions evaluated by valid-support coverage. Each scenario fixes the source image and action prompt, then scores repeated model rollouts by their terminal outcomes.

observation x , an action prompt specifying the action a , a finite set of valid terminal outcomes $\mathcal{Y}$, and, when available, a reference distribution q over $\mathcal{Y}$. The model is queried K times with the same (x, a) pair, and the resulting videos are mapped to terminal outcomes and aggregated into an empirical outcome distribution $\hat{p}_M$. Keeping the source image and action prompt fixed ensures that variation across rollouts reflects the model-induced future distribution rather than changes in the model inputs.

As shown in Fig. 2, PAWBench contains 50 scenarios spanning eight mechanism groups, divided into **PAW-Calibration** and **PAW-Coverage**. **PAW-Calibration** includes 25 scenarios whose valid terminal outcomes have analytically specified or symmetry-derived reference distributions, such as tossing, rotation, routing, and draw-style randomizers. It tests whether $\hat{p}_M$ approaches the reference distribution q , or instead concentrates probability mass on a biased subset of valid futures. **PAW-Coverage** includes 25 scenarios whose valid outcomes can be enumerated but whose probabilities cannot be reliably specified, including collision, stability, agent interaction, and material transition scenarios. It tests whether repeated rollouts recover the qualitatively distinct valid futures in $\mathcal{Y}$. Detailed benchmark composition and outcome-set cardinalities are summarized in Fig. 9.

This division separates two failures that single-sample evaluation cannot distinguish. PAW-Calibration measures probability misallocation: every rollout may show a plausible die roll or spinner motion, yet one outcome may occur far too often. PAW-Coverage measures missing outcomes: individual rollouts may look plausible even though some valid futures never appear. We therefore report calibration and coverage separately rather than combine them into a single score.

3.2 SCENARIO CURATION

PAWBench relies on repeated-rollout statistics, so each scenario must make the induced future distribution well defined. We therefore curate scenarios around three requirements. First, stochasticity must arise from a visible physical mechanism, rather than from ambiguity in the action prompt or hidden initial conditions. Second, the action prompt must specify one atomic intervention whose completion can be judged from the generated video. Third, the terminal outcomes must form a finite, visually distinguishable set so that rollouts can be consistently mapped into $\mathcal{Y}$. We manually curate and review each of the 50 scenarios against these requirements.

For each scenario, we generate one source image (Google DeepMind, 2025b; OpenAI, 2026b) to represent the initial observation and use one action prompt to specify the action across all rollouts. We list the valid terminal outcomes and specify when a rollout fails to complete the intended physical process. For PAW-Calibration, we also derive a reference distribution from analytic reasoning or physical symmetry. We review and finalize the source image, action prompt, outcome set,

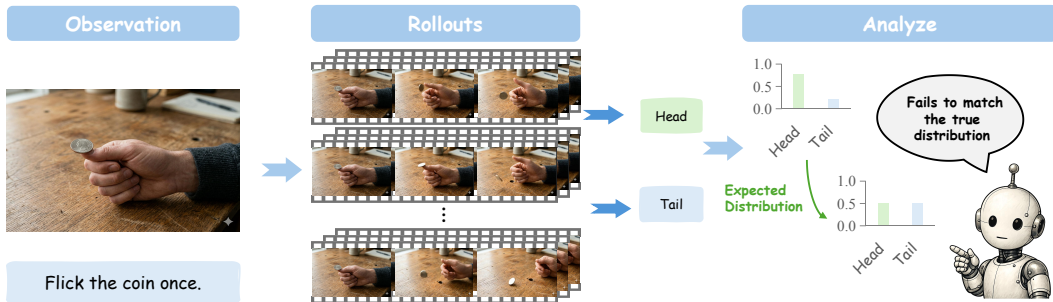


Figure 3: **PAWEval turns repeated rollouts into a distributional test.** In this PAW-Calibration example, a rubric-based judge maps each readable, in-schema coin-toss rollout to Head or Tail. Aggregating these labels yields the empirical outcome distribution, which is compared with the reference probabilities rather than matching generated videos frame by frame.

failure criteria, and reference distribution before evaluating any model. Appendix A.2 describes the construction process and the quality-control checks used to finalize each scenario.

3.3 EVALUATING DISTRIBUTIONS OVER POSSIBLE FUTURES

To estimate the distribution induced by a video generator, we introduce **PAWEval**, an outcome-level evaluation protocol for repeated rollouts. As illustrated in Fig. 3, we generate K video rollouts under the same (x, a) pair. Each scenario has an outcome rubric that specifies which terminal physical result to identify, how valid results map to $\mathcal{Y}$, and when to return an outcome-readout failure. Gemini 3.5 Flash (Google DeepMind, 2026) applies this rubric to each rollout and returns either a terminal-outcome label in $\mathcal{Y}$ or an outcome-readout failure. We normalize the assigned outcome labels to obtain the conditional empirical distribution $\hat{p}_M$; readout failures are reported separately because they do not represent physical outcomes in $\mathcal{Y}$ and cannot be counted as additional possible futures.

PAW-Calibration compares this conditional distribution with the reference distribution q using total variation distance (TVD), half the ℓ_1 distance between the two categorical distributions (Gibbs & Su, 2002). PAW-Coverage instead measures valid-support recovery as the fraction of valid outcomes observed across repeated rollouts. Both metrics use readable, in-schema outcomes; the outcome-readout gate separately determines whether a scene is scoreable. Appendix B provides the full metric definitions, gate, and aggregation rules used to compute the reported results.

4 EVALUATION ON PAWBENCH

4.1 EVALUATION SETUP

We evaluate eleven current video generation models on both PAW-Calibration and PAW-Coverage, as listed in Tab. 1. The roster spans proprietary and openly released systems: HappyHorse (Alibaba Cloud, 2026a), Veo3.1 Fast (Google, 2025), Kling 3 Std. (Kling AI, 2026), Seedance 2 (ByteDance Seed, 2026a), Wan2.7 (Alibaba Cloud, 2026b), Wan2.2 (Wan Team, 2025), LTX-2.3 (Lightricks, 2026b), LTX-2.5 (HaCohen et al., 2026; Lightricks, 2026a), Cosmos 3 Super I2V (NVIDIA, 2025; Agarwal et al., 2026), LingBot-Video-MoE (Ma et al., 2026), and MiniMax H3 (MiniMax, 2026). For each PAWBench scenario, all systems receive the same source image and action prompt. In the main evaluation, we sample $K = 50$ independent rollouts per scenario and system, with the action prompt held fixed across rollouts. Unless otherwise specified, we use each system with its default inference configuration. A scene passes the outcome-readout gate when no more than 30 of its 50 rollouts fail outcome readout, equivalently when at least 20 yield readable, in-schema outcomes. We report Scene Pass Rate (SPR) as the percentage of the 25 scenes in each track that pass this gate.

Table 1: **Main PAWBench results.** PAW-Calibration reports conditional TVD ($\times 100$, $\downarrow$): a 70/30 model distribution against a 50/50 reference scores 20, while an exact match scores 0. PAW-Coverage reports valid-support recovery ($\%$, $\uparrow$). Avg. is computed over passing scenes; SPR is the percentage of all 25 scenes that pass. and indicate the best and second-best conditional scores; / indicates that no scene in the corresponding mechanism group passes the gate.

Model	PAW-Calibration: TVD $\times 100$ ($\downarrow$)						PAW-Coverage: Coverage (%) ($\uparrow$)					
	Avg.	SPR	Toss	Rot.	Rout.	Draw	Avg.	SPR	Coll.	Stab.	Agent	Mat.
HappyHorse	43.1	92.0%	40.1	27.5	41.9	59.5	47.1	100.0%	45.0	36.8	49.0	58.3
Veo3.1 Fast	35.4	88.0%	37.1	12.2	32.4	55.0	41.8	100.0%	39.8	21.1	53.3	44.2
Kling 3 Std.	34.9	92.0%	29.9	20.3	38.7	46.0	52.8	88.0%	52.7	40.0	44.3	77.5
Seedance 2	30.5	100.0%	32.7	15.1	31.1	40.5	50.9	84.0%	48.6	50.0	44.3	67.5
Wan2.7	26.3	92.0%	26.0	12.2	26.8	36.5	50.0	96.0%	60.3	50.5	37.9	53.3
Wan2.2	26.3	64.0%	22.4	15.8	23.2	45.9	63.4	92.0%	76.8	56.7	55.0	58.3
LTX-2.3	30.1	24.0%	29.3	11.0	/	41.0	71.7	72.0%	82.7	50.0	59.5	90.0
LTX-2.5	30.2	60.0%	30.2	17.1	39.9	27.1	57.4	88.0%	66.7	30.0	59.4	57.5
Cosmos 3 Super I2V	20.5	80.0%	19.4	3.6	17.7	35.5	55.2	92.0%	56.7	31.7	49.4	81.7
LingBot-Video-MoE	41.8	72.0%	31.8	13.4	44.4	60.8	58.8	100.0%	62.2	61.8	46.5	72.5
MiniMax H3	24.2	68.0%	22.0	14.0	28.9	26.5	48.7	92.0%	59.5	39.5	36.4	58.3

4.2 BENCHMARK RESULTS

Current video generators remain far from probabilistically aligned world modeling. As shown in Tab. 1, no model combines well-aligned probability mass, broad support recovery, and reliable performance across scenes. Cosmos 3 Super I2V achieves the lowest Calibration TVD, but only 80.0% of its Calibration scenes pass the outcome-readout gate. LTX-2.3 attains the highest Coverage average, yet that average is computed over the 72.0% of its Coverage scenes that pass. Conversely, Seedance 2 and LingBot-Video-MoE pass every scene on Calibration and Coverage, respectively, but neither leads the corresponding conditional metric. Conditional alignment and scene-level reliability therefore capture distinct limitations of current models and must be read together.

Recovering valid futures is not the same as assigning them the right probability. The models that recover broad support are not those that best match the reference probabilities, and this separation persists across physical mechanisms. A generator may expose many plausible outcomes while allocating probability mass incorrectly, or closely match the frequencies of observed outcomes while leaving valid alternatives unseen. PAW-Coverage and PAW-Calibration therefore expose complementary failures: recovering the valid support is necessary, but it does not establish that probability mass is correctly allocated among the outcomes within that support.

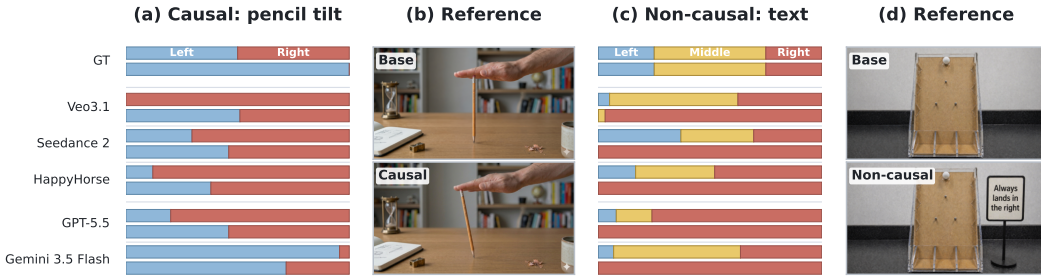


Figure 4: **Models underreact to physically causal interventions and overreact to non-causal cues.** Upper and lower bars show outcome distributions before and after intervention; panels (b) and (d) show the paired scenes. The pencil tilt is causal because it changes the physical transition, whereas the Galton-board text is non-causal because it leaves the transition unchanged.

Models Do Not Consistently Distinguish Causal from Non-Causal Changes. A probabilistic world model should change its future distribution only when the physical transition changes. We test this criterion with paired interventions: physically causal interventions alter the transition and its reference distribution, whereas non-causal interventions change only an irrelevant visual or textual signal and should therefore leave both the physical process and its future distribution unchanged.

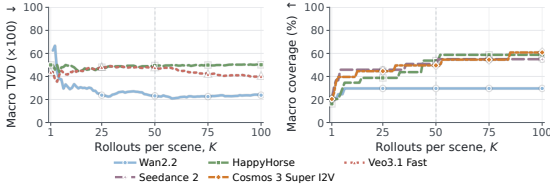


Figure 5: **Larger rollout budgets increase coverage but leave calibration largely unchanged.** Conditional TVD (left) and valid-support coverage (right) as the rollout budget increases from $K = 1$ to 100.

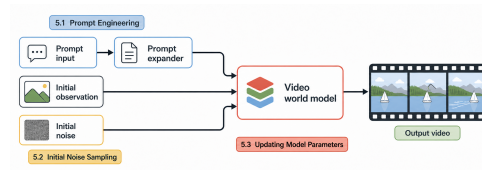


Figure 6: **Three interfaces for shaping a world model’s outcome distribution.** Section 5 studies prompt engineering, initial noise sampling, and updating model parameters.

As shown in Fig. 4, model distributions shift incompletely or in the wrong direction under physically causal interventions. Under non-causal interventions, distractor text redirects probability mass despite an unchanged reference distribution. Appendix B.5 reports the same pattern across all paired controls, video generators, and direct VLM future samplers. Models respond to altered inputs, but their responses do not reliably track whether the underlying physical transition has changed.

4.3 ROBUSTNESS OF THE EVALUATION PROTOCOL

The main results show that current models do not reproduce the relative frequencies of valid futures and recover only part of the valid outcome support. We test whether either finding reflects a limitation of the evaluation protocol: too few rollouts or disagreement between PAWEval and human judgments.

Rollout-Budget Sensitivity. Figure 5 shows that doubling the rollout budget leaves calibration largely unchanged, although coverage continues to rise for three of the four models. Additional samples can uncover more valid outcomes without correcting their observed relative frequencies. We therefore use $K = 50$ as a shared evaluation budget without treating it as a convergence point. The Monte Carlo analysis in Appendix B.4 tests whether finite sampling could still account for the calibration gap. Across the eleven video generators, observed TVD averages 31.2. For comparison, we draw matched samples from the reference distributions while preserving each model’s passing scenes and readable sample counts. In 99% of these simulations, average TVD remains below 9.22, far below the calibration error observed across the evaluated video generators.

Agreement with Human Judgments. We collect seven independent human judgments for each video using the same scene-specific outcome space. For the 888 videos where both PAWEval and the human panel provide a clear terminal-outcome label, PAWEval agrees with the decisive human label on 722 (81.3%). Disagreement over clear outcomes therefore cannot by itself account for the PAWBench gap. Appendix C provides the full protocol and analysis.

5 TOWARD PROBABILISTICALLY ALIGNED WORLD MODELING

Generating one plausible future, or even controlling which future is produced, does not amount to controlling a distribution of futures. A probabilistically aligned world model must instead preserve the stochastic structure of possible outcomes under a fixed initial observation and action. For a video generator used as a world model, this distribution can be influenced at three levels: **prompt engineering**, **initial noise sampling**, and **updating model parameters**, as summarized in Fig. 6. We therefore intervene at three interfaces that can shape this distribution: language specifies a requested future, initial noise sampling selects among futures, and model learning determines how probability mass is allocated among them. These probes distinguish inference-time steering and finite-budget exploration from changes in how the model itself distributes probability mass over future outcomes.

5.1 PROMPT ENGINEERING

Language can steer a video generator by naming a future in its prompt. If a model can already render several plausible futures, varying these requests across rollouts could reshape its output distribution. We therefore evaluate both requirements: whether the language controller selects futures with the required frequencies or support, and whether the generator realizes each selected future.

Table 2: **VLM distributions over possible futures.** Repeated VLM responses are mapped to PAWBench outcomes and aggregated into empirical distributions without video generation.

Model	PAW-Calibration: TVD $\times 100$ ($\downarrow$)						PAW-Coverage: Coverage (%) ($\uparrow$)					
	Avg.	SPR	Toss	Rot.	Rout.	Draw	Avg.	SPR	Coll.	Stab.	Agent	Mat.
Qwen3.5 Plus	40.9	84.0%	36.4	15.1	44.9	59.0	35.9	96.0%	19.8	49.5	48.6	29.2
GPT-5.5	42.3	100.0%	35.6	42.9	41.9	56.0	34.3	100.0%	24.9	29.6	44.7	39.2
GLM-5V Turbo	34.8	88.0%	24.1	36.9	33.9	50.8	39.9	100.0%	30.9	56.6	42.3	38.3
Kimi K2.6	38.2	96.0%	32.6	32.1	36.8	58.5	45.1	100.0%	44.1	55.7	46.5	34.2
Gemini 3.5 Flash	39.4	100.0%	37.0	29.0	40.6	52.0	46.6	96.0%	49.5	39.5	48.9	43.3

Table 3: **Predicted outcomes do not substitute for target outcomes.** The first row scores the outcomes selected by GPT-5.5 PE before video synthesis. Matched Base, PE, and Oracle PE results for four generators follow; all conditions cover 25 scenes per PAWBench track at $K = 50$.

Model	PAW-Calibration: TVD $\times 100$ ($\downarrow$)						PAW-Coverage: Coverage (%) ($\uparrow$)					
	Avg.	SPR	Toss	Rot.	Rout.	Draw	Avg.	SPR	Coll.	Stab.	Agent	Mat.
GPT-5.5 PE	44.3	100.0%	40.1	49.5	43.6	49.0	35.0	100.0%	35.4	34.6	35.5	33.3
Wan2.2	26.3	64.0%	22.4	15.8	23.2	45.9	63.4	92.0%	76.8	56.7	55.0	58.3
+PE	27.1	92.0%	23.5	15.4	28.8	39.5	61.9	100.0%	68.1	50.7	62.9	57.5
+Oracle PE	15.3	84.0%	10.5	13.9	14.0	27.3	76.2	96.0%	71.1	72.7	82.8	76.7
Cosmos 3 Super I2V	20.5	80.0%	19.4	3.6	17.7	35.5	55.2	92.0%	56.7	31.7	49.4	81.7
+PE	31.6	92.0%	27.0	15.0	34.9	47.0	64.9	100.0%	71.1	62.0	53.5	76.7
+Oracle PE	12.8	88.0%	17.6	9.2	13.2	5.1	87.3	100.0%	84.8	91.4	91.3	80.8
MiniMax H3	24.2	68.0%	22.0	14.0	28.9	26.5	48.7	92.0%	59.5	39.5	36.4	58.3
+PE	38.9	96.0%	32.5	35.0	42.2	48.6	49.8	96.0%	59.3	41.8	40.5	57.5
+Oracle PE	10.8	96.0%	21.6	6.0	2.1	12.6	82.9	96.0%	82.4	91.4	82.1	76.7
LTX-2.5	30.2	60.0%	30.2	17.1	39.9	27.1	57.4	88.0%	66.7	30.0	59.4	57.5
+PE	36.4	96.0%	34.3	28.7	36.6	48.0	55.6	100.0%	59.8	42.1	51.4	68.3
+Oracle PE	18.0	88.0%	26.2	21.8	15.0	5.7	73.9	96.0%	56.0	78.0	89.9	73.3

We test three settings. First, prior work finds that language models often understand or describe a target distribution more accurately than they reproduce it through repeated sampling, although distribution-aware prompting and training can narrow this gap (Meister et al., 2025; Gu et al., 2025; Misaki & Akiba, 2025; Sorensen et al., 2026). We therefore ask whether VLMs can act as samplers over possible futures without being given the target probabilities. We repeatedly query five VLMs (Qwen Team, 2026; OpenAI, 2026a; Z.AI, 2026; Moonshot AI, 2026; Google DeepMind, 2026) under the same initial observation and action, map each prediction to the PAWBench outcome space, and aggregate the predictions into an empirical distribution. Second, GPT-5.5 performs prompt engineering (PE) for each rollout: without access to the reference distribution, it predicts a possible outcome and writes a generator prompt requesting that outcome. Third, for Oracle PE, we directly specify one target outcome in each generator prompt, arranging the targets across rollouts to follow the reference distribution in PAW-Calibration and to balance the valid outcomes in PAW-Coverage. Together, these settings separate two sources of error: predicting the wrong distribution of outcomes and failing to produce a requested outcome in video. Direct VLM future sampling tests the first, PE evaluates the full chain, and Oracle PE supplies the target outcomes directly to isolate the second.

The distributions produced by direct VLM sampling are already misaligned. Tab. 2 shows that GLM-5V Turbo achieves the lowest Calibration TVD at 34.8, while Gemini 3.5 Flash achieves the highest Coverage at 46.6%. The outcomes selected by GPT-5.5 for PE are also misaligned before video generation, scoring 44.3 Calibration TVD and 35.0% Coverage in Tab. 3. Passing these selections to video generators raises SPR for all four models, but increases Calibration TVD among passing scenes in every case and improves Coverage for only two. Manually specified targets perform better: Oracle PE lowers Calibration TVD and raises Coverage for every generator. Even with these targets, Fig. 7 shows that the generators realize only 37.6–58.1% of the requested outcomes. Current language-based control is limited by

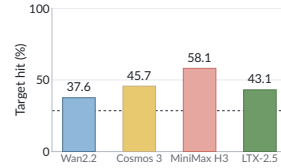


Figure 7: **Oracle PE often misses requested outcomes.**

both errors: the controller selects the wrong distribution of futures, and the generators often miss the supplied target. It can change individual rollouts without reliably aligning their distribution.

5.2 INITIAL NOISE SAMPLING

The language interventions change which future is requested, but leave open what can be recovered when the request itself is held fixed. We therefore turn from language to sampling: noise determines which future is reached under the same request. Holding the action prompt and generator fixed, we test whether part of the observed gap is a finite-sample exploration failure, in which independent draws repeatedly visit the same modes even when other valid outcomes remain accessible.

We adopt Couple to Control (C2C) (Jia et al., 2026), a repulsive Gaussian coupling scheme that introduces negative dependence among the $K = 50$ initial-noise samples while preserving each sample’s standard Gaussian marginal. The action prompt, generator, and rollout budget remain fixed across C2C and independent sampling; Appendix D.2 details the coupled-noise construction and the matched evaluation conditions used to isolate the effect of noise coupling.

Table 4: **Coupled noise broadens finite-gallery exploration.** C2C (Jia et al., 2026) couples $K = 50$ rollout noises while preserving the standard Gaussian marginal of each noise.

Model	PAW-Calibration: TVD $\times 100$ ($\downarrow$)						PAW-Coverage: Coverage (%) ($\uparrow$)					
	Avg.	SPR	Toss	Rot.	Rout.	Draw	Avg.	SPR	Coll.	Stab.	Agent	Mat.
Wan2.2	26.3	64.0%	22.4	15.8	23.2	45.9	63.4	92.0%	76.8	56.7	55.0	58.3
+C2C	25.7	84.0%	17.4	22.3	21.6	48.7	69.2	88.0%	82.4	57.5	59.4	68.3
LTX-2.3	30.1	24.0%	29.3	11.0	/	41.0	71.7	72.0%	82.7	50.0	59.5	90.0
+C2C	19.9	32.0%	18.6	7.2	34.0	27.7	74.8	72.0%	92.8	87.5	57.1	76.7
Cosmos 3 Super I2V	20.5	80.0%	19.4	3.6	17.7	35.5	55.2	92.0%	56.7	31.7	49.4	81.7
+C2C	19.4	72.0%	25.8	6.1	14.3	24.2	63.9	92.0%	74.9	56.7	49.1	76.7

Tab. 4 reports results across all 25 scenes in each PAWBench track. Across the scenes that pass in each condition, C2C lowers mean Calibration TVD and raises mean Coverage for all three generators. The gains vary across mechanisms and do not consistently raise SPR. C2C therefore helps the 50 rollouts explore the model’s existing possibilities more broadly, rather than changing the distribution learned by the model. We therefore turn next to the model itself.

5.3 UPDATING MODEL PARAMETERS

Unlike language and noise interventions, changing the training distribution can alter the generator’s learned allocation of probability mass. We train five LoRA-adapted Wan2.2 models (Hu et al., 2022) on training sets with different ratios of left- and right-falling pencil videos, with the left-fall share ranging from 0% to 100%, while holding the training budget and recipe fixed. Each model is evaluated on upright and left-leaning pencil scenes using a direction-neutral action prompt and $K = 50$ rollouts. Because the interior mixtures contain both outcomes, differences among them probe relative mass rather than support. Appendix D.3 provides the dataset, adaptation, and evaluation details.

Table 5: **Training mixtures reshape outcome mass.** TVD $\times 100$ is reported for the unadapted Base and five LoRA models trained on increasing proportions of left-fall examples.

Scene	Reference L/R	Base	LoRA training $P(\text{left})$ (%)				
			0	20	50	80	100
Upright	50/50	17.3	50.0	17.6	23.5	48.0	50.0
Left-leaning	100/0	41.3	97.2	50.0	18.9	0.0	0.0

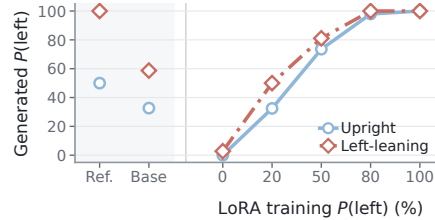


Figure 8: **Training steers outcome mass across scenes.** Both scenes follow similar trends.

Fig. 8 shows that increasing the share of left-falling videos in the training set raises the generated left-fall frequency in both scenes. The relationship is nonlinear: the generated frequency does not follow

the training proportion one-to-one. The consistent shift shows that training-data composition can reshape the model’s outcome distribution; imbalanced outcome frequencies may therefore contribute to current generators’ distributional biases, although their actual training distributions are unknown. A probabilistically aligned model should produce a 50/50 distribution for the upright pencil and a 100/0 distribution for the left-leaning pencil. None of the five adapted models produces both (Tab. 5). With 20% left-falling videos in training, the model comes closest to the upright reference, while the left-leaning scene remains at 50/50. Increasing that share to 80% or 100% brings the left-leaning scene to its 100/0 reference, but also makes the upright pencil fall left almost every time. The same adjustment moves both scenes in the same direction, so improving the match for one scene worsens the other. Changing global outcome frequencies therefore provides only coarse control over the model’s distribution. Probabilistic alignment requires learning how the distribution of possible futures should change with each scene’s initial physical state under a fixed action.

6 RELATED WORK

Video Generators as World Models. World models have long been used to support planning by predicting the consequences of actions, from classical model-based reinforcement learning to latent dynamics and decision-centric models (Sutton, 1990; Ha & Schmidhuber, 2018; Hafner et al., 2020; Schrittwieser et al., 2020). Recent video systems increasingly pursue this role through controllable visual rollouts, including large-scale video generators and interactive world models (OpenAI, 2024; Bruce et al., 2024; Google DeepMind, 2025a; NVIDIA, 2025; Yang et al., 2024). This shift makes repeated rollouts from the same initial observation and action natural. PAWBench focuses on the distributional question left open by these systems: not only whether a rollout is coherent, controllable, or visually realistic, but whether the induced samples place probability mass on the right futures.

Video-generation and world-model benchmarks. Video-generation benchmarks have substantially improved evaluation of visual quality, temporal consistency, and text-video alignment (Liu et al., 2024; Huang et al., 2024a;b). Physics-focused benchmarks further probe physical plausibility and action following (Bansal et al., 2024; Meng et al., 2024; Guo et al., 2025; Sanli et al., 2025; Pu et al., 2026; Li et al., 2026). Recent world-model benchmarks evaluate action-conditioned prediction, physical-law adherence, embodied consistency, or downstream utility for planning and control (Li et al., 2025; Qin et al., 2025; Kang et al., 2024; Tian et al., 2023). These evaluations are complementary to PAWBench, but their unit of analysis is usually an individual generation, a deterministic violation, or task success under a prompt. PAWBench instead holds the initial observation and action fixed, samples repeated rollouts, maps them to terminal outcome labels, and evaluates whether the empirical outcome distribution or support matches the stochastic structure of the scene.

Calibration and coverage under fixed actions. Calibration and distributional evaluation separate accuracy, fidelity, probability assignment, and support recovery (Guo et al., 2017; Nalisnick et al., 2019; Sajjadi et al., 2018; Kynkäänniemi et al., 2019; Naeem et al., 2020). Recent work on generative-video uncertainty also shows that confidence estimation is feasible and important (Mei et al., 2025a;b). Closest to our setting, CaliBench evaluates nine stochastic scenes with known reference distributions, reporting scorability separately from conditional TVD and testing for miscalibration (Sadeghi et al., 2026). PAWBench shares the repeated-rollout, discrete-outcome perspective but separates two regimes: PAW-Calibration when defensible reference probabilities are available, and PAW-Coverage when only the valid support can be specified. It further pairs these measurements with controlled probes of causal state, language, sampling noise, and training distribution.

7 LIMITATIONS

PAWBench is designed as a diagnostic test of probabilistic alignment, and its current form has three main limitations. First, it evaluates stochastic futures through terminal outcomes, which makes distributional comparison tractable but does not fully capture trajectory-level dynamics or intermediate physical processes. Second, its estimates are based on a finite number of rollouts: larger sampling budgets can reveal the induced distribution more reliably, but they increase evaluation cost and do not by themselves correct biased model distributions. Third, PAWBench uses controlled, visually parseable scenarios to isolate stochastic future modeling, leaving longer-horizon, interactive,

and embodied environments for future study. Future work should extend probabilistic alignment from terminal labels to richer state trajectories, study more efficient and reliable rollout-based estimators, scale the benchmark to interactive settings, and develop training objectives for models that explicitly learn calibrated distributions over possible futures across different physical states.

8 CONCLUSION

World modeling requires more than producing plausible continuations; a world model should capture the distribution of possible futures under the same initial observation and action. We introduce PAWBench to make this requirement measurable through repeated video rollouts. Across eleven current video generators, no model consistently matches the reference probabilities while recovering the range of valid futures, and the gap cannot be explained by finite sampling or disagreement with human judgments on clear outcomes. Controlled interventions further show that model distributions do not reliably track causal changes in the physical process. Language can request individual futures, coupled noise can broaden finite-budget exploration, and fine-tuning can shift outcome frequencies, but none reliably recovers the scene-conditioned distribution over possible futures. PAWBench therefore shows that plausible, diverse, or controllable rollouts do not by themselves establish probabilistically aligned world modeling across different physical states.

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. URL <https://arxiv.org/abs/2606.02800>.
- Alibaba Cloud. Alibaba rolls out HappyHorse 1.0 in limited beta, 2026a. URL https://www.alibabacloud.com/blog/alibaba-rolls-out-happyhorse-1-0-in-limited-beta_603068.
- Alibaba Cloud. Wan2.7, 2026b. URL <https://www.alibabacloud.com/help/en/model-studio/video-generate-edit-model>. Model Studio documentation.
- Mohammad Babaeizadeh, Chelsea Finn, Dumitru Erhan, Roy H. Campbell, and Sergey Levine. Stochastic variational video prediction. In *International Conference on Learning Representations*, 2018. URL <https://research.google/pubs/stochastic-variational-video-prediction/>.
- Hritik Bansal, Zongyu Lin, Tianyi Xie, Zeshun Zong, Michal Yarom, Yonatan Bitton, et al. VideoPhy: Evaluating physical commonsense for video generation. *arXiv preprint arXiv:2406.03520*, 2024. URL <https://arxiv.org/abs/2406.03520>.
- Jake Bruce, Michael Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, et al. Genie: Generative interactive environments. *arXiv preprint arXiv:2402.15391*, 2024. URL <https://arxiv.org/abs/2402.15391>.
- ByteDance Seed. Seedance 2.0, 2026a. URL https://seed.bytedance.com/en/seedance2_0.
- ByteDance Seed. Seedance 2.5, 2026b. URL <https://seed.bytedance.com/en/blog/one-take-creation-flexible-referencing-introducing-seedance-2-5>.
- Emily Denton and Rob Fergus. Stochastic video generation with a learned prior. In *Proceedings of the 35th International Conference on Machine Learning*, 2018. URL <https://proceedings.mlr.press/v80/denton18a.html>.
- Alison L. Gibbs and Francis Edward Su. On choosing and bounding probability metrics. *International Statistical Review*, 70(3):419–435, 2002. doi: 10.1111/j.1751-5823.2002.tb00178.x.
- Google. Veo 3.1 Fast, 2025. URL <https://ai.google.dev/gemini-api/docs/video>. Gemini API documentation.
- Google DeepMind. Genie 3: A new frontier for world models, 2025a. URL <https://deepmind.google/blog/genie-3-a-new-frontier-for-world-models/>.
- Google DeepMind. Nano banana pro, 2025b. URL <https://deepmind.google/models/gemini-image/pro/>.
- Google DeepMind. Gemini 3.5 flash, 2026. URL <https://deepmind.google/models/model-cards/gemini-3-5-flash/>.
- Jia Gu, Liang Pang, Huawei Shen, and Xueqi Cheng. Do LLMs play dice? exploring probability distribution sampling in large language models for behavioral simulation. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 5375–5390, 2025. URL <https://aclanthology.org/2025.coling-main.360/>.
- Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, 2017. URL <https://arxiv.org/abs/1706.04599>.
- Xuyang Guo, Jiayan Huo, Zhenmei Shi, Zhao Song, Jiahao Zhang, and Jiale Zhao. T2VPhysBench: A first-principles benchmark for physical consistency in text-to-video generation. *arXiv preprint arXiv:2505.00337*, 2025. URL <https://arxiv.org/abs/2505.00337>.

- David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018. URL <https://arxiv.org/abs/1803.10122>.
- Yoav HaCohen, Benny Brazowski, Nisan Chiprut, Yaki Bitterman, Andrew Kvochko, Avishai Berkowitz, Daniel Shalem, Daphna Lifschitz, Dudu Moshe, Eitan Porat, Eitan Richardson, Guy Shiran, Itay Chachy, Jonathan Chetboun, Michael Finkelson, Michael Kupchick, Nir Zabari, Nitzan Guetta, Noa Kotler, Ofir Bibi, Ori Gordon, Poriya Panet, Roi Benita, Shahar Armon, Victor Kulikov, Yaron Inger, Yonatan Shiftan, Zeev Melumian, and Zeev Farbman. LTX-2: Efficient Joint Audio-Visual Foundation Model. *arXiv preprint arXiv:2601.03233*, 2026. URL <https://arxiv.org/abs/2601.03233>.
- Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations*, 2020. URL <https://research.google/pubs/dream-to-control-learning-behaviors-by-latent-imagination/>.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL <https://arxiv.org/abs/2106.09685>.
- Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, et al. VBench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024a. URL https://openaccess.thecvf.com/content/CVPR2024/papers/Huang_VBench_Comprehensive_Benchmark_Suite_for_Video_Generative_Models_CVPR_2024_paper.pdf.
- Ziqi Huang, Fan Zhang, Xiaojie Xu, Yinan He, Jiashuo Yu, Ziyue Dong, et al. VBench++: Comprehensive and versatile benchmark suite for video generative models. *arXiv preprint arXiv:2411.13503*, 2024b. URL <https://arxiv.org/abs/2411.13503>.
- Jing Jia, Liyue Shen, and Guanyang Wang. Couple to control: Joint initial noise design in diffusion models. *arXiv preprint arXiv:2605.11311*, 2026. URL <https://arxiv.org/abs/2605.11311>.
- Bingyi Kang, Yang Yue, Rui Lu, Zhijie Lin, Yang Zhao, Kaixin Wang, Gao Huang, and Jiashi Feng. How far is video generation from world model: A physical law perspective. *arXiv preprint arXiv:2411.02385*, 2024. URL <https://arxiv.org/abs/2411.02385>.
- Kling AI. Kling Video 3.0 Model User Guide, 2026. URL <https://kling.ai/quickstart/klingai-video-3-model-user-guide>.
- Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. In *Advances in Neural Information Processing Systems*, 2019. URL <https://arxiv.org/abs/1904.06991>.
- Dacheng Li, Yunhao Fang, Yukang Chen, Shuo Yang, Shiyi Cao, Justin Wong, Michael Luo, Xiaolong Wang, Hongxu Yin, Joseph E. Gonzalez, Ion Stoica, Song Han, and Yao Lu. WorldModelBench: Judging video generation models as world models. In *Advances in Neural Information Processing Systems*, volume 38. Curran Associates, Inc., 2025. doi: 10.52202/085713-1834. URL https://papers.nips.cc/paper_files/paper/2025/hash/4ec03ed08a3fcb59e1c815b5598beff1-Abstract-Datasets_and_Benchmarks_Track.html. Datasets and Benchmarks Track.
- Fei-Fei Li and World Labs Team. A functional taxonomy of world models: Renderers, simulators, planners, and the loop that connects them, June 2026. URL <https://www.worldlabs.ai/blog/taxonomy-of-world-models>.
- Shiyang Li, Yuewen Cao, Yihao Liu, Yuandong Pu, Baochang Zhang, Xiaofei Li, and Changqing Zou. Acoustitrace: When plausible sound violates physics, 2026. URL <https://arxiv.org/abs/2608.02035>.

- Lightricks. LTX-2.5, 2026a. URL <https://ltx.io/model/ltx-2-5>.
- Lightricks. LTX-2.3 Video Engine, 2026b. URL <https://ltx.io/model/ltx-2-3>.
- Yaofang Liu, Xiaodong Cun, Xuebo Liu, Xintao Wang, Yong Zhang, Haoxin Chen, Yang Liu, Tiejong Zeng, Raymond Chan, and Ying Shan. Evalcrafter: Benchmarking and evaluating large video generation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22139–22149, June 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Liu_EvalCrafter_Benchmarking_and_Evaluating_Large_Video_Generation_Models_CVPR_2024_paper.html.
- Shuailei Ma, Jiaqi Liao, Xinyang Wang, Jingjing Wang, Chaoran Feng, et al. Scaling mixture-of-experts video pretraining for embodied intelligence. *arXiv preprint arXiv:2607.07675*, 2026. URL <https://arxiv.org/abs/2607.07675>.
- Zhiting Mei, Ola Shorinwa, and Anirudha Majumdar. How confident are video models? empowering video models to express their uncertainty. *arXiv preprint arXiv:2510.02571*, 2025a. URL <https://arxiv.org/abs/2510.02571>.
- Zhiting Mei, Tenny Yin, Micah Baker, Ola Shorinwa, and Anirudha Majumdar. World models that know when they don’t know: Controllable video generation with calibrated uncertainty. *arXiv preprint arXiv:2512.05927*, 2025b. URL <https://arxiv.org/abs/2512.05927>.
- Nicole Meister, Carlos Guestrin, and Tatsunori Hashimoto. Benchmarking distributional alignment of large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 24–49, 2025. URL <https://aclanthology.org/2025.naacl-long.2/>.
- Fanqing Meng, Jiaqi Liao, Xinyu Tan, Wenqi Shao, Quanfeng Lu, Kaipeng Zhang, Yu Cheng, Ping Luo, et al. Towards world simulator: Crafting physical commonsense-based benchmark for video generation. *arXiv preprint arXiv:2410.05363*, 2024. URL <https://arxiv.org/abs/2410.05363>.
- MiniMax. MiniMax H3: An open model breaking the boundaries between tasks and modalities, July 2026. URL <https://www.minimax.io/blog/minimax-h3>.
- Kou Misaki and Takuya Akiba. String seed of thought: Prompting llms for distribution-faithful and diverse generation. *arXiv preprint arXiv:2510.21150*, 2025. URL <https://arxiv.org/abs/2510.21150>.
- Moonshot AI. Kimi-k2.6, 2026. URL <https://huggingface.co/moonshotai/Kimi-K2.6>.
- Muhammad Ferjad Naeem, Achkan Bhalgat, Vladislav Golyanik, Christian Theobalt, and Hans-Peter Seidel. Reliable fidelity and diversity metrics for generative models. In *Proceedings of the 37th International Conference on Machine Learning*, 2020. URL <https://arxiv.org/abs/2002.09797>.
- Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Gorur, and Frank Wood. Do deep generative models know what they don’t know? In *International Conference on Learning Representations*, 2019. URL <https://arxiv.org/abs/1810.09136>.
- NVIDIA. Cosmos world foundation model platform for physical AI. *arXiv preprint arXiv:2501.03575*, 2025. URL <https://arxiv.org/abs/2501.03575>.
- OpenAI. Video generation models as world simulators, 2024. URL <https://openai.com/index/video-generation-models-as-world-simulators/>.
- OpenAI. Gpt-5.5 model, 2026a. URL <https://developers.openai.com/api/docs/models/gpt-5.5>.
- OpenAI. Gpt image 2, April 2026b. URL <https://developers.openai.com/api/docs/models/gpt-image-2>.

- Yuandong Pu, Le Zhuo, Songhao Han, Jinbo Xing, Kaiwen Zhu, Shuo Cao, Bin Fu, Si Liu, Hongsheng Li, Yu Qiao, Wenlong Zhang, Xi Chen, and Yihao Liu. Picabench: How far are we from physically realistic image editing?, 2026. URL <https://arxiv.org/abs/2510.17681>.
- Yiran Qin, Zhelun Shi, Jiwen Yu, Xijun Wang, Enshen Zhou, Lijun Li, Zhenfei Yin, Xihui Liu, Lu Sheng, Jing Shao, Lei Bai, and Ruimao Zhang. WorldSimBench: Towards video generation models as world simulators. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 50338–50362. PMLR, 13–19 Jul 2025. URL <https://proceedings.mlr.press/v267/qin25f.html>.
- Qwen Team. Qwen3.5: Towards native multimodal agents, 2026. URL <https://qwen.ai/blog?id=qwen3.5>.
- Sahithya Ravi, Aditya Chinchure, Pushkar Shukla, Vered Shwartz, and Leonid Sigal. Position: World models must live in parallel worlds. In *NeurIPS 2025 Workshop on Bridging Language, Agent, and World Models for Reasoning and Planning*, 2025. URL <https://openreview.net/forum?id=sj9Nyke43S>.
- Jonathan Sadeghi, Jenny Seidenschwarz, Jesse Allardice, Sirish Srinivasan, Benjamin Graham, and Jeffrey Hawke. CaliBench: Are the stochastic dynamics of video world models physically calibrated? *arXiv preprint arXiv:2608.16829*, 2026. URL <https://arxiv.org/abs/2608.16829>.
- Mehdi S. M. Sajjadi, Olivier Bachem, Mario Lucic, Olivier Bousquet, and Sylvain Gelly. Assessing generative models via precision and recall. In *Advances in Neural Information Processing Systems*, 2018. URL <https://arxiv.org/abs/1806.00035>.
- Enes Sanli, Baris Sarper Tezcan, Aykut Erdem, and Erkut Erdem. Can your model separate yolks with a water bottle? Benchmarking physical commonsense understanding in video generation models. *arXiv preprint arXiv:2507.15824*, 2025. URL <https://arxiv.org/abs/2507.15824>.
- Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, Timothy Lillicrap, and David Silver. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588:604–609, 2020. URL <https://www.nature.com/articles/s41586-020-03051-4>.
- Taylor Sorensen, Benjamin Newman, Jared Moore, Chan Young Park, Jillian Fisher, Niloofar Mireshghallah, Liwei Jiang, and Yejin Choi. Spectrum tuning: Post-training for distributional coverage and in-context steerability. In *International Conference on Learning Representations*, 2026. URL <https://openreview.net/forum?id=ulvp7cbZeU>.
- Richard S. Sutton. Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In *Proceedings of the Seventh International Conference on Machine Learning*, 1990. URL https://papers.nips.cc/paper_files/paper/1990/hash/d9fc5b73a8d78fad3d6dffe419384e70-Abstract.html.
- Stephen Tian, Chelsea Finn, and Jiajun Wu. A control-centric benchmark for video prediction. *arXiv preprint arXiv:2304.13723*, 2023. URL <https://arxiv.org/abs/2304.13723>.
- Wan Team. Wan2.2, 2025. URL <https://github.com/Wan-Video/Wan2.2>.
- Sherry Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Leslie Kaelbling, Dale Schuurmans, and Pieter Abbeel. Unisim: Learning interactive real-world simulators. In *International Conference on Learning Representations*, 2024. URL <https://arxiv.org/abs/2310.06114>.
- Z.AI. Glm-5v-turbo overview, 2026. URL <https://docs.z.ai/guides/vlm/glm-5v-turbo>.

A BENCHMARK DETAILS

A.1 SCENARIO DEFINITION AND TASK TAXONOMY

Each PAWBench scenario fixes the inputs and outcomes of one repeated-rollout test, as shown in Tab. 6. A source image represents the initial observation, an action prompt specifies one atomic intervention, and a finite outcome set $\mathcal{Y}$ lists the valid terminal outcomes of the physical process. Each scenario also includes outcome-readout criteria and, when it can be derived independently of model outputs, a reference distribution q over $\mathcal{Y}$. We finalize these elements before evaluating any model.

A.1.1 PAW-CALIBRATION TASKS

PAW-Calibration comprises scenarios whose reference distribution q is fixed before model evaluation. We derive q from sector proportions, combinatorial outcome counts, physical symmetry, or visible causal state—not from model outputs or a default uniform assumption. When the initial observation and action do not support a defensible reference distribution, the scenario is assigned to PAW-Coverage or discarded. PAW-Calibration tests whether repeated rollouts allocate probability mass in accordance with q .

A.1.2 PAW-COVERAGE TASKS

PAW-Coverage comprises scenarios whose valid terminal outcomes can be enumerated but whose relative probabilities cannot be justified from the initial observation and action. It tests support recovery across repeated rollouts without assuming equal outcome probabilities.

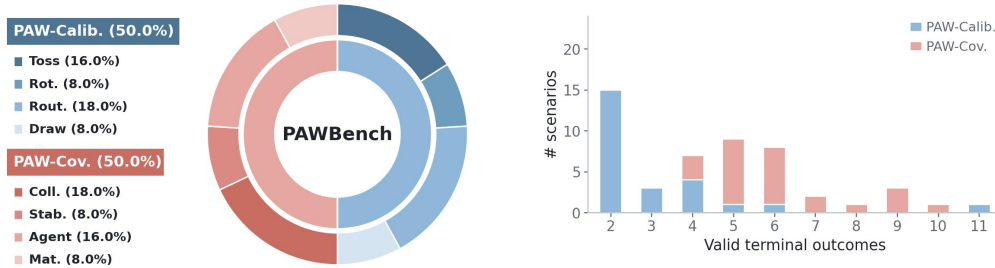


Figure 9: **Benchmark statistics of PAWBench.** PAWBench contains 50 manually curated scenarios, evenly divided between PAW-Calibration and PAW-Coverage, across eight stochastic mechanism groups. Each scenario fixes one source image and action prompt and uses a compact terminal-outcome set, allowing repeated rollouts to be aggregated into an empirical outcome distribution.

A.2 SCENARIO CONSTRUCTION AND QUALITY REVIEW

We construct each scenario so that variation across repeated rollouts reflects the model’s future distribution rather than ambiguity in the benchmark item. Before evaluation, we finalize the source image, action prompt, valid terminal outcomes, reference distribution when applicable, and outcome-readout criteria. The physical mechanism must be visible, the action must specify one intervention, and the terminal outcomes must be distinguishable.

Source-image selection. For each scenario, we generate candidate source images with image generation models (Google DeepMind, 2025b; OpenAI, 2026b) and manually select one image. The selected image must clearly show the relevant physical mechanism and action target, avoid hidden initial conditions, and support terminal outcomes that can be distinguished from the final frames. We reject candidates with an obscured mechanism, an ambiguous initial state, or competing action targets.

Task family	Task definition	Terminal readout	Representative audit concerns
PAW-Calibration			
Tossing	An object is thrown, flicked, or released into one of several settled states.	Face, orientation, or resting region.	No throw or release; continuity break; settled state unreadable.
Rotation	An object spins relative to a fixed partition, pointer, or orientation frame.	Final sector or orientation.	No rotation; reference frame or endpoint unreadable; out-of-schema orientation.
Routing	A released object traverses a visible branching structure.	Terminal branch, lane, bin, pocket, or location.	No traversal; continuity break; terminal location unreadable or out of schema.
Draw	One item is selected or released from a visible finite collection.	Selected identity or category.	No selection; item continuity break; selected identity unreadable or out of schema.
PAW-Coverage			
Collision	An object undergoes contact or a target-directed interaction.	Contact, deflection, scoring, miss, or settling state.	Intervention absent; impossible transition; continuity or readability failure.
Stability	A near-threshold arrangement is subjected to a perturbation.	Stable, shifted, toppled, or collapsed state.	No perturbation; continuity break; terminal state unreadable or out of schema.
Agent interaction	A living agent receives a fixed visible stimulus.	Observable response category.	Stimulus not delivered as specified; agent continuity break; response unreadable.
Material transition	A material or deformable object undergoes a fixed intervention.	Resulting physical or material state.	Intervention absent; material continuity break; final state unreadable or out of schema.

Table 6: **PAWBench task taxonomy.** Each family identifies a recurring physical task, its terminal outcome readout, and representative outcome-readout or trustworthiness concerns. Outcome readout supplies the labels and scene gate used by PAWBench; trustworthiness checks are auxiliary diagnostics (Appendix B.1).

Scenario finalization. Given the selected source image, we write an action prompt that specifies one atomic intervention whose completion can be judged from the generated video. We then define the valid terminal outcomes, criteria for readable in-schema labels, and separate trustworthiness checks. For PAW-Calibration, we retain a reference distribution only when analytic reasoning or physical symmetry justifies it independently of model outputs. Outcome categories describe terminal physical states rather than incidental differences in appearance or intermediate trajectories.

Quality review. We review each candidate as a complete scenario before evaluating any model. The review checks mechanism visibility, action specificity, terminal-outcome distinguishability, reference-distribution justification when applicable, and the outcome-readout criteria. Failed checks lead us to select or generate a new source image, revise the action prompt, or refine the outcome categories. We discard a scenario when its ambiguity cannot be removed without changing the physical process being evaluated.

B EVALUATION PROTOCOL AND SUPPORTING ANALYSES

B.1 PAWEVAL OUTCOME READOUT

PAWEval uses Gemini 3.5 Flash (Google DeepMind, 2026) to apply a frozen scene-specific outcome rubric to each generated rollout. The rubric returns a terminal label $y \in \mathcal{Y}$ when the endpoint is readable and belongs to the outcome set; otherwise, it returns the shared readout-failure label $\perp$. When a visible endpoint is more specific than the outcome categories, the rubric maps it to the corresponding category before aggregation. These outcome labels determine conditional TVD and

Coverage. Readout failures instead determine whether a scene passes the outcome-readout gate and therefore affect SPR; they never enter the conditional outcome distribution.

We also run an auxiliary trustworthiness audit that records whether the requested action is completed, object continuity is preserved, and the transition follows the specified physical mechanism. This separate diagnostic does not change the terminal label or any PAWBench score. Tab. 7 summarizes the scoring readout and the auxiliary audit.

For example, in a routing scene whose schema contains only named terminal bins, a ball that comes to rest visibly outside those bins has a readable but out-of-schema endpoint and is therefore recorded as an outcome-readout failure. By contrast, a spinner that ends clearly on blue after an implausible or discontinuous rotation still receives the in-schema label blue, while the trajectory is flagged only by the separate trustworthiness audit and the blue label still enters the conditional outcome counts.

Criterion	Required condition	Failure condition
Outcome readout		
Endpoint readability	The terminal state can be determined visually.	The endpoint is hidden, ambiguous, or unreadable.
Schema membership	The terminal state maps to the fixed outcome set $\mathcal{Y}$.	A readable endpoint falls outside $\mathcal{Y}$.
Trustworthiness audit		
Action completion	The requested intervention is executed as specified.	The action is absent or the clip is off task.
Object continuity	Key objects remain identifiable throughout the rollout.	Objects disappear, duplicate, or change identity.
Physical process	The transition follows the specified physical mechanism.	The transition violates the specified physical mechanism.

Table 7: **PAWEval separates scoring from auxiliary diagnostics.** Outcome readout supplies the labels and scene gate used by PAWBench; trustworthiness criteria only diagnose how readable outcomes are realized.

Outcome-readout failures remain separate from the conditional distribution over readable, in-schema outcomes. Fig. 10 shows the fixed PAWEval prompt scaffold into which the scene-specific outcome and trustworthiness rubrics are inserted.

B.2 AUXILIARY TRUSTWORTHINESS DIAGNOSTICS

The trustworthiness audit asks how a readable endpoint was reached. A clip may reach such an endpoint without completing the intended physical process: the action may not be executed, object identity may break, or the endpoint may be reached through physically implausible dynamics. These flags do not change the outcome label, TVD, Coverage, or SPR.

The audit covers three types of failure: action execution, physical process, and object continuity. Fig. 11 shows that physical-process errors account for the largest share overall, while action-execution and object-continuity errors remain substantial across mechanism groups. The figure therefore describes how readable endpoints are reached; it does not change the scores in Tab. 1.

B.3 METRIC COMPUTATION AND AGGREGATION

PAWBench first applies a scene-level outcome-readout gate. For each model–scene pair, we evaluate $K = 50$ rollouts generated under the same initial observation and action. The outcome readout assigns each rollout either a terminal label $y_i \in \mathcal{Y}$ or the readout-failure label $\perp$. Let m denote the number of rollouts assigned to $\perp$. A scene passes the gate when $m \leq 30$, equivalently when at least 20 rollouts have readable, in-schema outcomes. Otherwise, the scene does not contribute a conditional alignment score.

Prompt for PAWEval Rubric Judging

Fixed two-axis judge scaffold

System: You are a strict PAWEval JSON judge. Return one valid JSON object and no prose.

Runtime inputs shared by both axes:

- Scene: {scene_id}; Sample: {sample_id}
- Evidence: {evidence_summary}
- Scene rubric: {rubric_text}
- Video evidence: sampled frames attached in initial / action / terminal order

Axis 1: Outcome readout

PAWEval Outcome Readout

Use the provided evidence and scene rubric to return one outcome label.

Do not judge trustworthiness in this axis.

Return a single JSON object only. Do not use Markdown fences, explanations, or any text before or after the JSON object. The response must begin with { and end with }.

Return JSON with this exact shape and preserve the provided scene_id:

```
{
  "scene_id": "{scene_id}",
  "outcome_readable": true,
  "outcome_in_schema": "IN_SCHEMA",
  "outcome_label": "<one rubric label or null>",
  "observed_result": "<short evidence-grounded observation>",
  "failure_axes": []
}
```

If the result is not readable, set outcome_readable=false, outcome_in_schema="NOT_READABLE", outcome_label=null, and include the reason in failure_axes. If the result is readable but outside the rubric, set outcome_in_schema="OUT_OF_SCHEMA" and outcome_label=null.

Runner addendum: append JSON schema paweval.outcome_readout/v1, require one canonical rubric label for IN_SCHEMA, and end with "Return JSON only."

Axis 2: Trustworthiness audit

PAWEval Trustworthiness Audit

Audit whether the evidence supports strict reporting. Do not choose the outcome label in this axis. Return a single JSON object only.

The top-level status field must be exactly one of: TRUSTED, QUESTIONABLE, UNTRUSTED, UNCERTAIN.

Do not use PASS or FAIL as the top-level status. PASS, FAIL, UNCLEAR, and NOT_APPLICABLE are allowed only inside nested audit blocks such as scene_grounding.status and action_execution.status.

Return JSON with this exact shape and preserve the provided scene_id:

```
{
  "scene_id": "{scene_id}",
  "status": "TRUSTED",
  "scene_grounding": {"status": "PASS", "notes": ""},
  "action_execution": {
    "status": "PASS",
    "target_hit": "YES",
    "object_acquired": "YES",
    "action_spec_followed": "YES",
    "notes": ""
  },
  "object_continuity": {"status": "PASS", "notes": ""},
  "physical_process": {"status": "PASS", "notes": ""},
  "failure_axes": []
}
```

Use TRUSTED only when all required evidence supports the report. Use QUESTIONABLE for incomplete or ambiguous evidence, UNTRUSTED for contradictory evidence, and UNCERTAIN when the evidence cannot support a reliable audit.

For this automated runner, keep every notes field as the empty string "".

Name concrete failure axes in failure_axes rather than adding prose outside the JSON object.

Runner addendum: append JSON schema paweval.trustworthiness_audit/v1, require the official trust enums, and end with "Return JSON only."

Figure 10: **PAWEval prompt scaffold.** The scene-specific outcome rubric assigns a terminal label or an outcome-readout failure. A separate trustworthiness rubric checks action execution, physical process, and object continuity without changing the PAWBench score. The placeholders {evidence_summary} and {rubric_text} are filled with sampled-frame evidence and the scene-specific rubric.

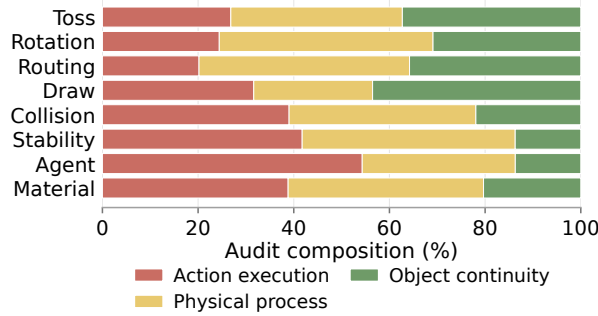


Figure 11: **Failure anatomy.** Incidence composition by mechanism group for eight models: Happy-Horse, Veo 3.1 Fast, Kling 3 Std., Seedance 2, Wan2.7, Wan2.2, LTX-2.3, and Cosmos 3 Super I2V. Percentages use classifiable audits and count non-exclusive failures.

Conditional metrics are computed only for passing scenes. Let $n_{\text{readout}} = K - m$. The empirical distribution over readable, in-schema outcomes is

$$\hat{p}_{M,\text{readout}}(y) = \frac{1}{n_{\text{readout}}} \sum_{i=1}^K \mathbb{1}[y_i = y], \quad y \in \mathcal{Y}.$$

The readout-failure label $\perp$ is not included in this conditional distribution.

For PAW-Calibration, each scenario specifies a reference distribution q over its valid terminal outcomes. We measure valid-only conditional total variation distance,

$$\text{TVD}_{\text{readout}}(\hat{p}_{M,\text{readout}}, q) = \frac{1}{2} \sum_{y \in \mathcal{Y}} |\hat{p}_{M,\text{readout}}(y) - q(y)|.$$

Lower TVD indicates closer probability-mass alignment among readable, in-schema outcomes. We report $100 \times \text{TVD}$.

For PAW-Coverage, the valid outcome set is enumerable but its probabilities are not specified. We measure valid-support recovery,

$$\text{COV}_{\text{readout}}(\hat{p}_{M,\text{readout}}, \mathcal{Y}) = \frac{|\{y \in \mathcal{Y} : \hat{p}_{M,\text{readout}}(y) > 0\}|}{|\mathcal{Y}|}.$$

Higher coverage indicates that repeated rollouts recover a broader set of valid terminal outcomes. We report coverage as a percentage. Because observed support depends on the rollout budget, all primary comparisons fix $K = 50$; Fig. 5 examines sensitivity to increasing the budget to $K = 100$.

Mechanism-group scores average the scene-level conditional metric over passing scenes in that group. The track-level *Avg.* instead averages over all passing scenes in the corresponding 25-scene track; it is not the simple average of the four mechanism-group scores. Scene Pass Rate is $n_{\text{pass}}/25$, where the denominator includes the complete track roster. Conditional alignment scores and Scene Pass Rate should therefore be read together. PAW-Calibration and PAW-Coverage remain separate and are not combined into a single ranking.

B.4 FINITE-SAMPLE EFFECTS ON CALIBRATION

Because PAW-Calibration estimates an outcome distribution from a finite number of readable rollouts, even samples drawn from the reference distribution q can produce nonzero TVD. We measure this sampling error with matched Monte Carlo simulations. Models pass different sets of scenes and yield different numbers of readable rollouts, so we construct a separate matched baseline for each model. For each passing model–scene pair, we draw the same number of readable outcomes from the scene’s reference distribution and compute TVD using the same aggregation as in the main evaluation. These simulations preserve each model’s passing scenes and readable sample counts, replacing only the generated outcomes with samples from q .

Figure 12 compares the observed model TVDs with these matched baselines. Across the eleven video generators, the observed TVD averages 31.2. Samples drawn from the reference distributions average 8.33 TVD, and 99% of the simulated averages remain below 9.22. Each generator’s observed TVD also exceeds the 99th percentile of its own matched baseline. Finite sampling therefore contributes to measured TVD, but it is too small to explain the observed calibration gap. Table 8 shows that the same conclusion holds under alternative aggregation and a stricter minimum-readout requirement. This analysis concerns conditional Calibration TVD and does not evaluate Coverage or outcome-readout failures.

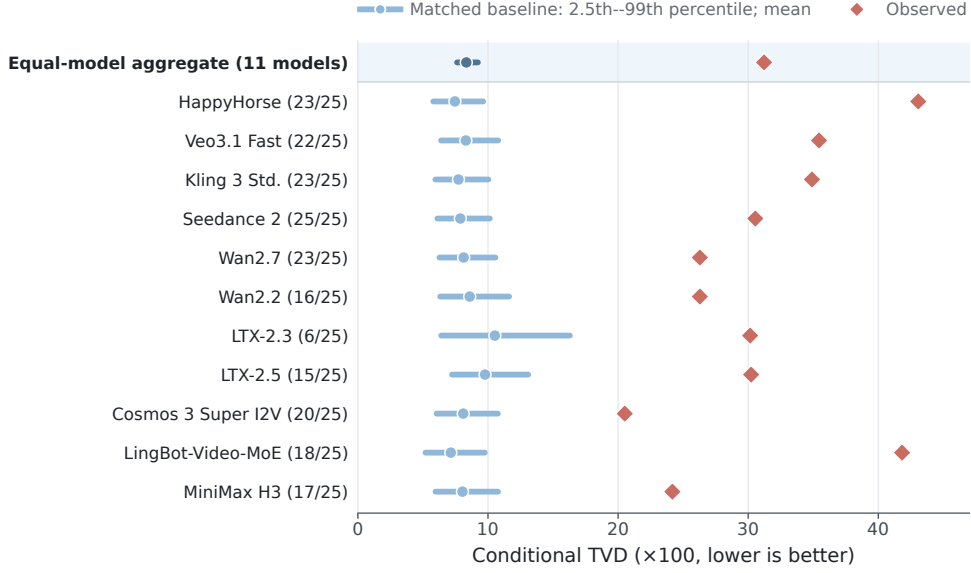


Figure 12: **Observed TVD exceeds the matched finite-sample baseline.** Blue intervals show the 2.5th to 99th percentiles obtained by sampling from the reference distributions with each model’s passing scenes and readable sample counts. Circles mark the simulated means, and red diamonds mark the observed model TVDs. The aggregate row first averages passing scenes within each model and then weights the eleven models equally.

Table 8: **Finite-sample null sensitivity.** We repeat the matched analysis with pooled passing cells and with a stricter $n_{\text{readout}} \geq 30$ threshold. The primary and stricter analyses weight models equally; the pooled analysis weights model–scene cells equally. All rows use $B = 50,000$ replicates, and p_{MC} is the plus-one upper-tail probability. These checks do not alter benchmark scores or model rankings.

Aggregation	Models	Cells	Observed	Null mean	Null q99	Exceedances	p_{MC}
Equal-model (primary)	11	208	31.2	8.33	9.22	0/50,000	2.0×10^{-5}
Pooled passing cells	11	208	31.6	8.13	8.90	0/50,000	2.0×10^{-5}
Equal-model, $n_{\text{readout}} \geq 30$	11	159	31.8	6.67	7.44	0/50,000	2.0×10^{-5}

B.5 FULL CAUSAL AND NON-CAUSAL CONTROLS

Figs. 13–16 expand the paired-control diagnostic in Fig. 4 to eight controlled pairs. The roster contains all eleven video generators from the main evaluation and five vision-language models that sample future outcomes directly from the same initial observation and action. For every system and scene, we draw $K = 50$ samples and project readable responses into the same fixed outcome set. Video-generator rows use PAWEval outcome readout on generated videos, whereas VLM rows use projected semantic predictions. The two groups therefore support a diagnostic comparison of paired response direction, not a cross-modality performance ranking.

Within each system, the upper bar shows the base scene and the lower bar shows its causal or non-causal variant. Bars report $P(\text{outcome} \mid \text{readable, in-schema})$ when the scene passes the same outcome-readout gate used in the main evaluation; gray bars marked / denote gate failures. A causal intervention changes the physical transition and reference distribution, so the model distribution should change accordingly. A non-causal intervention preserves both, so the distribution should remain stable.

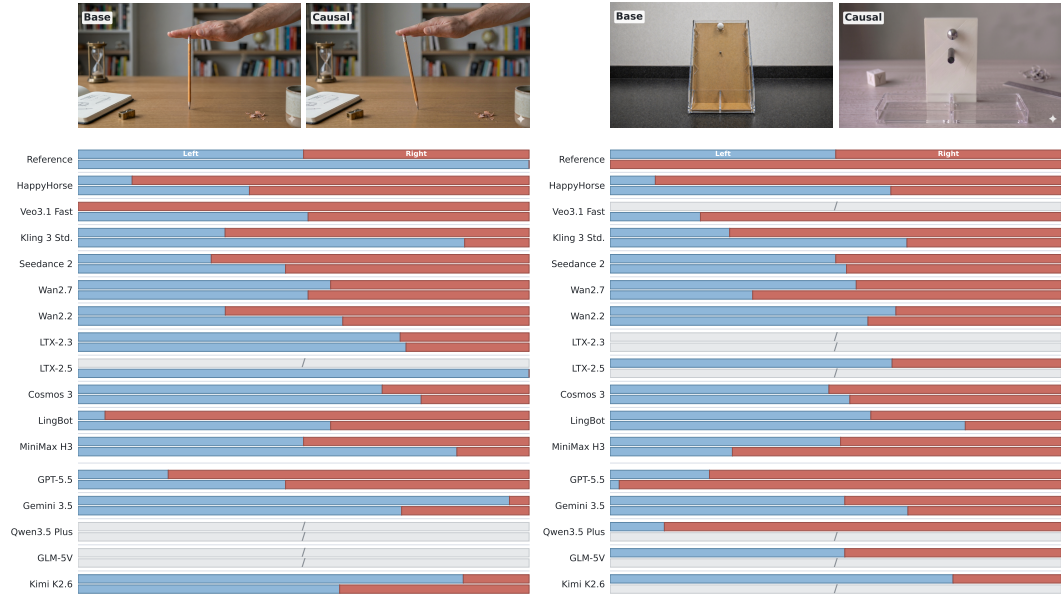


Figure 13: **Causal-state controls across video generators and direct VLM future samplers.** Each panel compares the outcome distribution for a base scene (upper bar) with that for a causal variant (lower bar), whose reference distribution changes. Bars report distributions conditional on readable, in-schema outcomes; gray bars marked / denote scenes that fail the outcome-readout gate.



Figure 14: **Non-causal routing controls.** The paired scenes change an irrelevant cue while preserving the physical transition and reference distribution. Upper and lower bars show the base and cue-perturbed outcome distributions, respectively; gray bars marked / denote scenes that fail the outcome-readout gate.

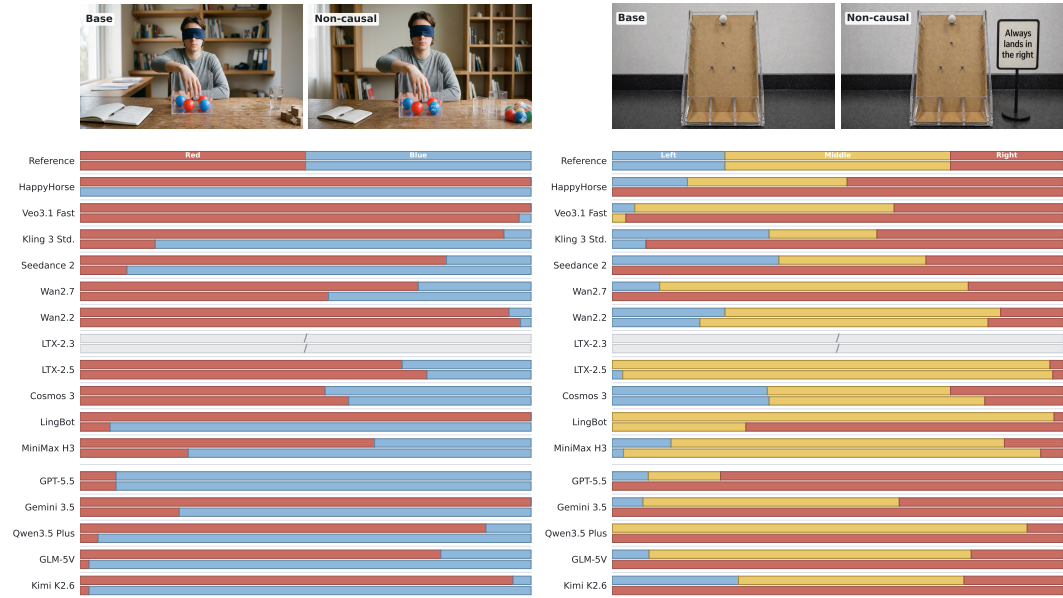


Figure 15: **Non-causal draw and text controls.** Distractor appearance in the blind-draw scene and an outcome-suggestive sign in the Galton-board scene leave the reference distribution unchanged. Several systems nevertheless move substantial probability mass between the paired conditions.

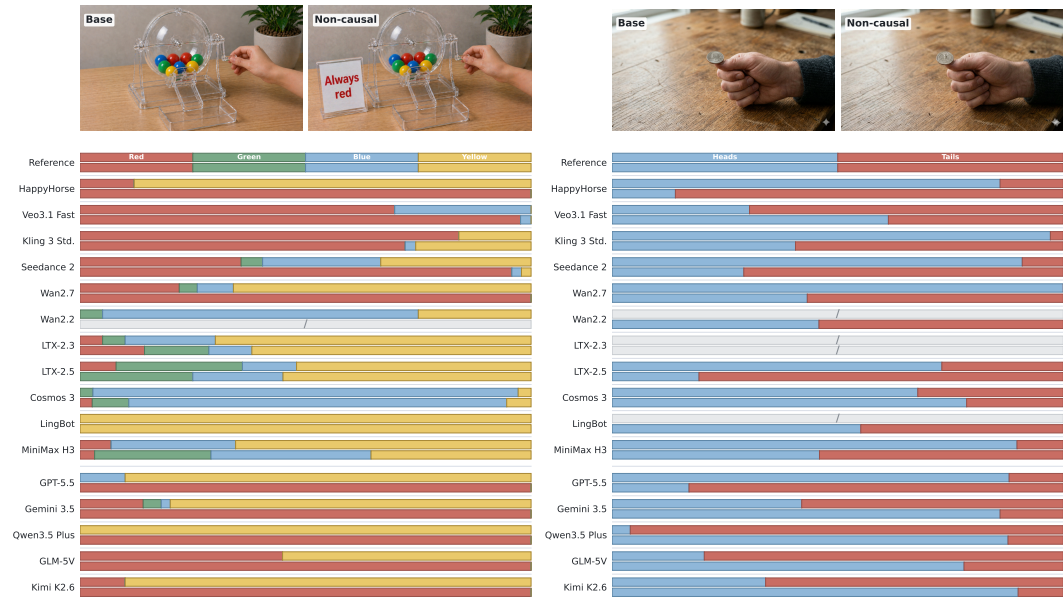


Figure 16: **Non-causal dispenser and coin controls.** Outcome-suggestive text or an irrelevant appearance change leaves the underlying chance process and reference distribution unchanged. Upper and lower bars show the base and cue-perturbed distributions conditional on readable, in-schema outcomes.

C HUMAN STUDY AND PAWEVAL ALIGNMENT

The human study asks whether PAWEval and a human panel assign the same terminal outcome to the same generated video. It audits PAWEval’s outcome readout; it does not validate PAWBench’s TVD, Coverage, or model rankings.

Study sampling. The frozen study frame contains 1,500 generated videos from 29 PAWBench scenes. Of these, 1,200 were selected to balance scene coverage, 150 to strengthen within-scene model comparisons, and 150 to probe cases expected to be difficult for the evaluator. Each video is one study item and receives seven independent judgments from a screened Rapidata audience. The resulting agreement therefore characterizes this frozen study frame rather than all PAWBench rollouts.

Annotation and comparison protocol. Each Rapidata task¹ shows one generated video and asks annotators to identify its final visible outcome using a scene-specific closed-choice question. The options express the canonical PAWBench outcomes in human-readable terms and include *Cannot tell* for unclear, hidden, or unreadable endpoints. Annotators do not see PAWEval’s prediction or model metadata. We map their responses back to the canonical outcome space and define the human label as decisive when the seven votes have a unique modal outcome other than *Cannot tell*; ties and *Cannot tell* modes remain non-comparable. A PAWEval label is comparable when it is an in-schema physical outcome. Agreement is an exact canonical-label match on the intersection of these two conditions.

The two eligibility conditions are parallel filters over the same 1,500 videos: humans provide a decisive label on 1,128 videos, and PAWEval provides an in-schema label on 1,024. Their intersection contains 888 comparable videos.

Agreement analysis. Tab. 9 reports 81.3% exact agreement: PAWEval matches the decisive human label on 722 of the 888 comparable videos. The 888 videos are a subset of the full 1,500-video study frame, so this result is not an all-row accuracy estimate.

Descriptively, agreement rises with the strength of the human mode: 58.7% for three or four matching votes, 77.3% for five, and 91.7% for six or seven. This stratification shows where PAWEval–human agreement is concentrated; it does not extend the 81.3% result beyond the comparable subset. Overall, the study supports a bounded conclusion: PAWEval often agrees with human judgment when both produce clear terminal-outcome labels. PAWBench calibration and coverage remain defined by the repeated-rollout metrics in Appendix B.3.

Table 9: **PAWEval–human agreement on comparable videos.** Agreement is the exact canonical-label match when both PAWEval and the seven-vote human panel provide a clear physical outcome. Consensus rows are descriptive strata of the 888-video comparison set.

Human-consensus subset	Videos	Exact matches	Agreement
Overall	888	722	81.3%
Low consensus (3–4 votes)	213	125	58.7%
Moderate consensus (5 votes)	154	119	77.3%
High consensus (6–7 votes)	521	478	91.7%

D INTERVENTION EXPERIMENTS

Section 5 intervenes at three interfaces where future distributions may be shaped: the language supplied to the generator, the initial noise used to sample it, and the training distribution absorbed by the model. This appendix specifies how each intervention is constructed and which quantities are held fixed. Quantitative comparisons are reported beside their corresponding protocols.

¹<https://www.rapidata.ai/>

Table 10: **Experimental settings for the Section 5 probes.** Each row records the evaluated systems, prompt interface, sampling budget, and readout; the corresponding constructions follow below.

Probe	Systems	Prompt interface	Sampling	Readout and scope
VLM future sampling	Qwen3.5 Plus, GPT-5.5, GLM-5V Turbo, Kimi K2.6, Gemini 3.5 Flash	Fixed VLM scaffold; source image and action prompt are inserted at runtime	25 scenes per track; $K = 50$ queries; hosted-model randomness	Fixed GPT-5.5 projection; a scene fails above 30 unmapped or invalid responses
PE / Oracle PE	Wan2.2, Cosmos 3 Super I2V, MiniMax H3, LTX-2.5	GPT-5.5 predicts an outcome and writes the PE prompt; Oracle PE manually places scheduled targets in the prompts	25 scenes per track; $K = 50$ rollouts under each condition	PAWEval target hit plus Calibration TVD or Coverage; all current model-track pairs
C2C	Wan2.2, LTX-2.3, Cosmos 3 Super I2V	Base prompt unchanged	$G = 10$ groups of $m = 5$; matched $K = 50$ IID gallery	Common PAWEval protocol over all 25 scenes in both tracks
LoRA	Wan2.2 I2V-A14B with five rank-32 adaptations	Direction-neutral pencil prompt	$K = 50$ matched seeds for two physical states	PAWEval outcome counts; dose-response claims use adapted models, with Base as reference

D.1 LANGUAGE-SIDE DIAGNOSTICS AND INTERVENTIONS

We evaluate language at two interfaces: predicting possible outcomes without video synthesis and requesting a specific outcome from a video generator. In PE, GPT-5.5 predicts a possible outcome from the initial observation and action without access to the reference distribution, then writes a generator prompt that requests this outcome. In Oracle PE, we instead write the target outcome directly into the generator prompt. These settings separate two errors: predicting the wrong outcome distribution and failing to produce a requested outcome in video.

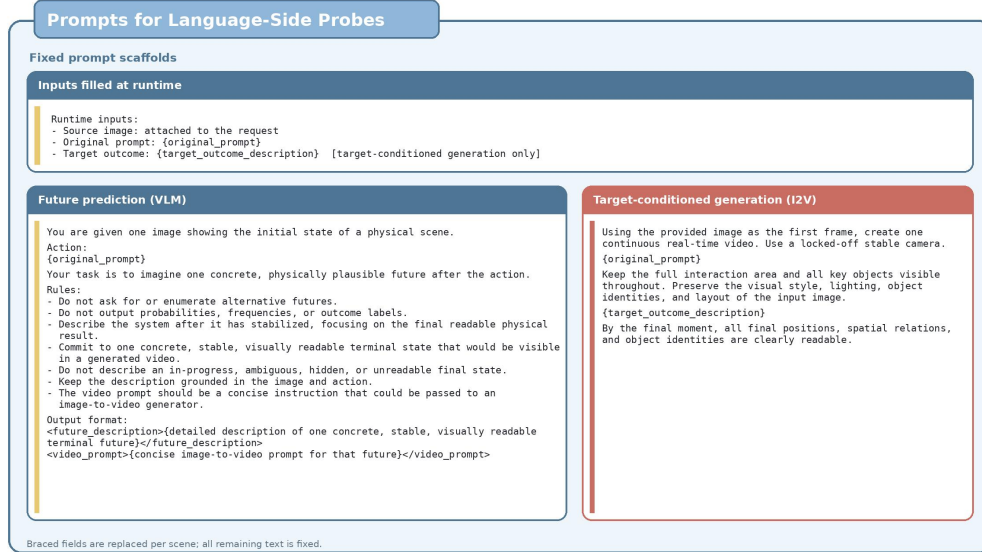


Figure 17: **Prompts for the two language-side probes.** The VLM scaffold requests one plausible future without exposing outcome labels or probabilities. The target-conditioned I2V scaffold instead adds one terminal outcome to the original scene prompt. Braced fields are filled at runtime.

VLM sampling without video. We first test whether repeated VLM predictions produce the required outcome distribution before video generation. For each of the 25 scenes in both tracks, we show a VLM the same source image and action prompt and ask it to describe one possible outcome, repeating the query $K = 50$ times. A fixed GPT-5.5 projector maps these free-form predictions to

the scene-specific PAWBench outcome set, yielding an empirical distribution that can be evaluated against the scene’s outcome specification.

Only responses mapped to valid outcomes contribute to this conditional distribution. Unmapped or invalid responses instead count toward the scene gate, which fails a scene when they exceed 30 of the 50 queries. Over passing scenes, Calibration reports valid-only TVD and Coverage reports support recovery; SPR separately gives the fraction of the 25 scenes that pass. We reuse the frozen responses and projections for both tracks rather than regenerating or reprojecting them.

Prompt Engineering(PE). For each rollout, GPT-5.5 predicts one possible outcome from the initial observation and action, then writes a generator prompt requesting that outcome. It receives neither the reference distribution nor a target outcome from the benchmark.

Oracle PE. For Oracle PE, we manually specify one PAWBench target outcome in every generator prompt. For PAW-Calibration, we convert the reference probabilities into $K = 50$ target requests using largest-remainder allocation. For PAW-Coverage, we divide the 50 requests as evenly as possible across the valid outcomes. We deterministically permute each target schedule across rollout indices and reuse the same schedule for every generator. Across the 25 scenes in each track, this gives 2,500 target-conditioned rollouts for each generator. PAWEval reads the terminal outcome, and a rollout counts as a target hit only when that outcome matches the request; unreadable, invalid, missing, and different outcomes are all misses. We then ask a separate distributional question by comparing the resulting Calibration TVD and Coverage with matched Base rollouts. Target-hit rate measures whether a generator follows an individual outcome request, whereas the PAWBench metrics measure the aggregate distribution produced across requests.

Tab. 3 reports Base, GPT-5.5 PE, and Oracle PE results for Wan2.2, Cosmos 3 Super I2V, MiniMax H3, and LTX-2.5. All conditions cover the 25 scenes in each track at $K = 50$.

Language steers outcomes but does not determine their distribution. Tab. 2 shows that repeated VLM predictions remain misaligned with the reference distribution. Tab. 3 shows that PE raises Calibration TVD relative to Base for all four generators and improves Coverage for only two, whereas Oracle PE lowers mean Calibration TVD and raises mean Coverage for all four. Fig. 7 nevertheless shows that only 37.6–58.1% of their rollouts reach the requested outcome. The results expose both errors: the language controller predicts the wrong outcome distribution, and the video generator often fails to produce a requested outcome. Language can change individual rollouts without reliably aligning their aggregate distribution.

D.2 COUPLED NOISE SAMPLING

The initial noise of a diffusion generator is usually sampled independently across rollouts. C2C asks a narrower question: can a finite gallery cover the generator’s accessible futures more evenly when these noises are coupled, while preserving the standard-Gaussian marginal seen by every rollout?

Marginal-preserving coupling. For group g , let $\epsilon_{g,1}, \dots, \epsilon_{g,m}$ be independent standard-Gaussian noise tensors. Following the centered construction of Couple to Control (Jia et al., 2026), we form

$$\epsilon_{g,i} \stackrel{\text{i.i.d.}}{\sim} \mathcal{N}(0, I), \quad \tilde{\epsilon}_{g,i} = \sqrt{\frac{m}{m-1}} \left(\epsilon_{g,i} - \frac{1}{m} \sum_{j=1}^m \epsilon_{g,j} \right). \quad (1)$$

Each $\tilde{\epsilon}_{g,i}$ remains marginally distributed as $\mathcal{N}(0, I)$, while the samples within a group satisfy $\sum_i \tilde{\epsilon}_{g,i} = 0$ and $\text{Cov}(\tilde{\epsilon}_{g,i}, \tilde{\epsilon}_{g,j}) = -I/(m-1)$ for $i \neq j$. Groups are constructed independently. The intervention therefore changes the joint distribution of the gallery without changing the noise distribution of any single rollout.

Experimental construction. We use $G = 10$ independent groups with $m = 5$ samples per group, yielding $K = Gm = 50$ rollouts for every model–scene pair. The coupled tensors are constructed at each generator’s native initial-latent shape and supplied at the start of denoising. The matched IID condition instead draws K independent tensors from the same $\mathcal{N}(0, I)$ marginal; the source image, prompt, generator, and remaining generation settings are held fixed. Tab. 4 reports Wan2.2, LTX-2.3, and Cosmos 3 Super I2V on the full 25-scene PAW-Calibration roster and the full 25-scene PAW-Coverage roster. All C2C and IID rollouts follow the common PAWEval outcome-readout and

aggregation protocol in Appendix B.3; the intervention changes only the dependence among initial noises, not the validity gate or metric denominator.

Because C2C preserves the marginal distribution of every initial noise, it does not alter the generator’s learned one-rollout conditional law. It instead probes whether a finite IID gallery misses futures that the generator can already realize. Improved coverage under coupling is therefore evidence about finite-budget exploration, not about learning a better-calibrated world model.

D.3 TRAINING-DISTRIBUTION INTERVENTION

The previous interventions leave the generator unchanged. Here we instead ask whether the relative frequency of two futures in the training data becomes part of the generator’s learned outcome distribution. We vary only the left–right composition of the training dataset, while holding its size, the underlying video pool, the training recipe, and the inference request fixed.

Training mixtures. Starting from 100 left-falling and 100 right-falling videos, we form five training datasets of 2,000 examples with left/right ratios of 0/100, 20/80, 50/50, 80/20, and 100/0. Source videos are repeated uniformly within each direction, so the datasets differ only in the relative frequency of the two outcomes. The three interior mixtures retain both futures and isolate their relative frequency; the two endpoints serve as boundary conditions. Every training example uses the same neutral motion prompt.

LoRA adaptation. Each mixture adapts the same Wan2.2 I2V-A14B base model with rank-32 LoRA (Hu et al., 2022). Training uses 832×480 clips of 49 frames, a learning rate of 10^{-4} , and 2,000 optimizer updates per expert; all other recipe choices are shared across mixtures. Because Wan2.2 divides denoising between high- and low-noise experts, we train paired adapters for each mixture and apply them to the corresponding experts at inference with unit LoRA weight. This produces five adapted models; the original Wan2.2 model provides an unadapted reference.

Evaluation across physical states. We evaluate Base and the five LoRA conditions on two pencil scenes using the same source image within each scene, a direction-neutral action prompt, and a matched set of $K = 50$ sampling seeds for every model–scene pair. The upright pencil has a 50/50 left–right reference, whereas the initially left-leaning pencil has a 100/0 reference. The five adapted models share one generation profile. The Base samples use the same scene, request, and seeds, but differ in output geometry and lack complete diffusion-profile metadata. We therefore draw the dose–response comparison from the LoRA models and use Base only to show the unadapted behavior. The two scenes test whether the intervention writes a shared directional prior or a distribution that changes with the observed physical state.

Table 11: **Outcome counts under the training-distribution intervention.** Each cell reports $L/R/\text{invalid}$ over $K = 50$; Fig. 8 plots $L/(L + R)$ over readable outcomes. Base denotes the unadapted model; dose–response comparisons use the LoRA rows.

Profile	Training L/R	Upright pencil	Left-leaning pencil
Base	–	16/33/1	27/19/4
LoRA	0/100	0/36/14	1/35/14
LoRA	20/80	12/25/13	17/17/16
LoRA	50/50	25/9/16	30/7/13
LoRA	80/20	48/1/1	50/0/0
LoRA	100/0	45/0/5	48/0/2

Across the three interior mixtures, conditional left-fall frequency increases in both scenes. Training composition therefore shifts mass among supported futures, but the experiment establishes neither exact ratio recovery nor scene-conditioned probability learning: both scenes respond similarly despite their different reference distributions.

E QUALITATIVE EXAMPLES AND FAILURE CASES

This appendix complements the aggregate PAWBench results with 13 case cards from distinct scenes. Each card shows a compact movie strip from one rollout, together with its scene, model, generation condition, instruction, and visible behavior. The examples illustrate how PAWEval reads terminal outcomes and why some visually plausible videos fail to complete the intended physical process. They provide qualitative evidence and do not enter any reported aggregate metric.

The set includes successful physical trials and typical failure modes: action-execution failures, object-continuity breaks, physically inconsistent trajectories, apparatus instability, and questionable but readable clips. These cards illustrate the aggregate patterns in Tab. 1, Fig. 11, Tab. 2, and Fig. 4.

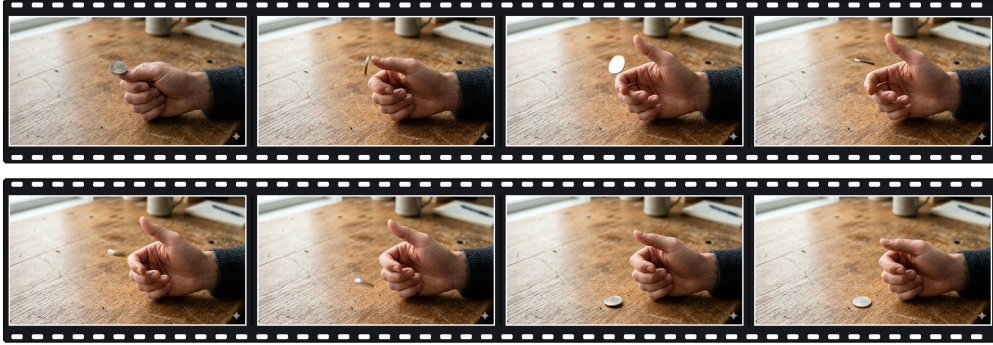


Figure 18: **Example of the scene “Coin flip” generated by HappyHorse.** The model is instructed to flick the coin once. The generated rollout ends with the coin lying heads-up on the table.

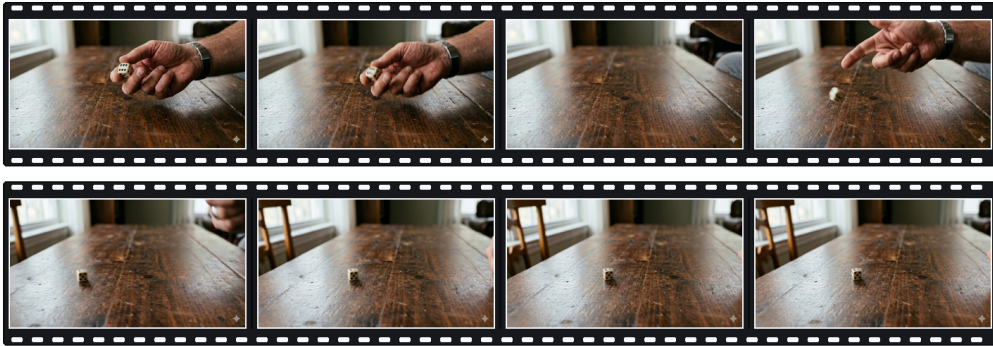


Figure 19: **Example of the scene “Die odd/even toss” generated by Kling 3 Std.** The model is instructed to throw the die once onto the table. The generated rollout ends with the die settled on the table, showing five pips on top.



Figure 20: **Example of the scene “Two-color equal-sector spinner” generated by Veo3.1 Fast.** The model is instructed to spin the wheel once under the fixed pointer. The final frame places the blue sector under the pointer, but the spin is visually inconsistent across the rollout.

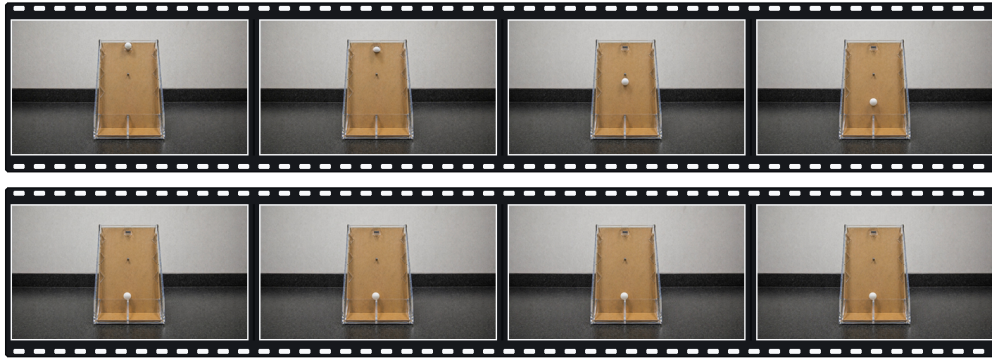


Figure 21: **Example of the scene “One-peg Galton board” generated by Wan2.2.** The model is instructed to release the ball once. The ball settles on the divider instead of entering either bin.

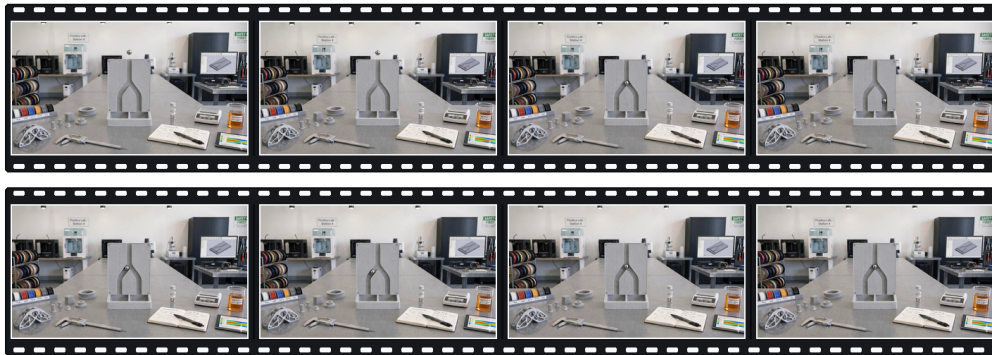


Figure 22: **Example of the scene “Y-track branch” generated by Seedance 2.** The model is instructed to release the ball once. The ball remains stuck at the split rather than traveling down one branch.

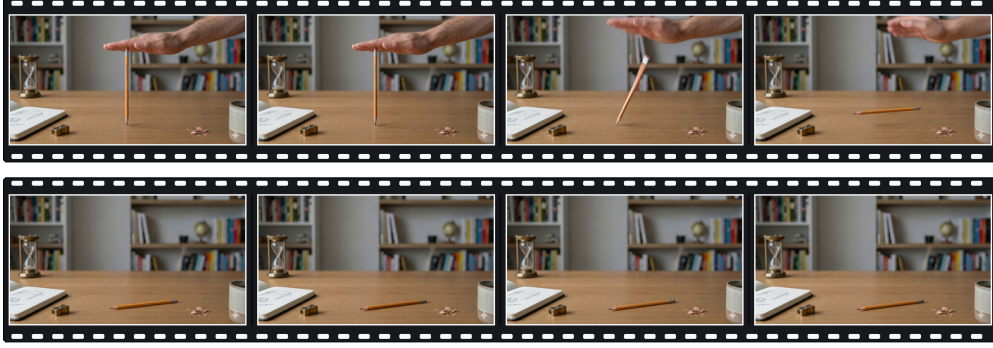


Figure 23: **Example of the scene “Vertical pencil fall” generated by Wan2.2.** The model is instructed to move the hand upward once and let the pencil fall. The generated rollout ends with the pencil falling to the right.



Figure 24: **Example of the scene “Blind ball draw” generated by Wan2.7.** The model is instructed to pick up exactly one ball once. The person selects a red ball and holds it up by the end of the rollout.

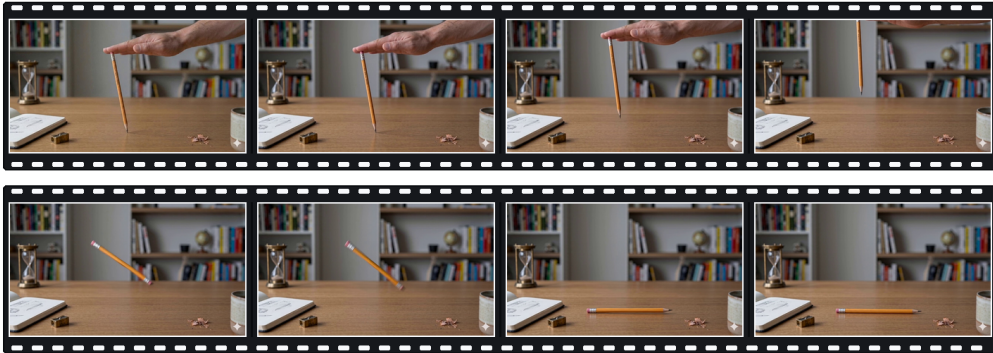


Figure 25: **Example of the scene “Left-Leaning Pencil Fall” generated by Seedance 2.** The model is instructed to move the hand upward once and let the pencil fall. The pencil appears to fall left, but its identity is not preserved cleanly through the motion.

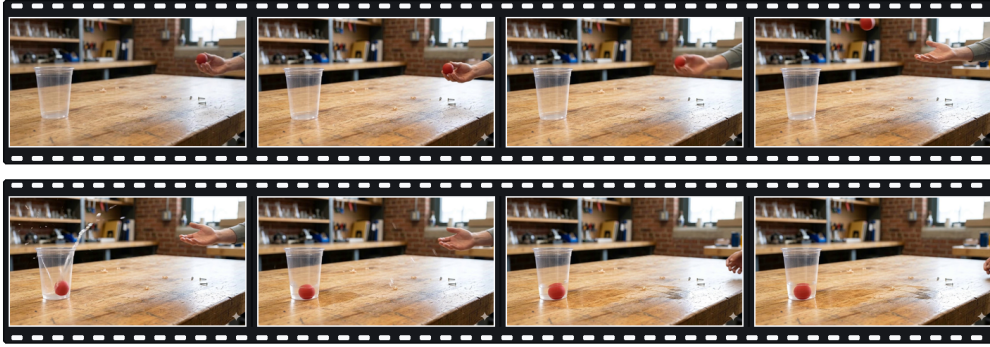


Figure 26: **Example of the scene “Ball Toss Into Cup” generated by HappyHorse.** The model is instructed to toss the ball once from the visible hand pose toward the cup. The generated rollout ends with the ball landing cleanly inside the cup.

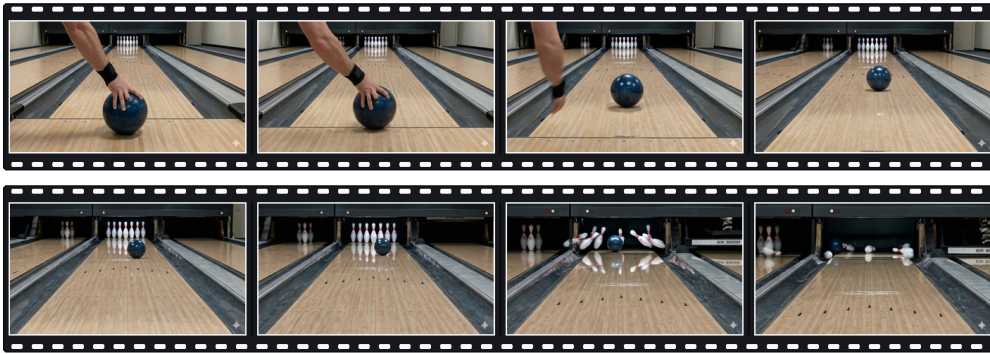


Figure 27: **Example of the scene “Seven-Pin Bowling Roll” generated by Kling 3 Std.** The model is instructed to roll the bowling ball once down the lane. The generated rollout ends with all seven pins knocked down.

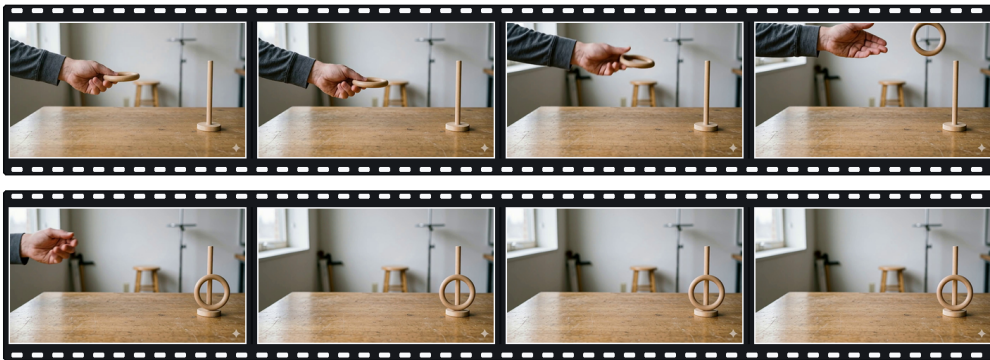


Figure 28: **Example of the scene “Ring toss toward peg” generated by MiniMax H3.** The model is instructed to toss the ring once from the visible hand pose toward the peg. The ring travels toward the peg and remains around it in the final frames.



Figure 29: **Example of the scene “Loose-yarn pull” generated by LTX-2.5 Base.** The model is instructed to pull the loose yarn end once. The loose yarn is pulled upward and visibly extends from the knitted fabric.

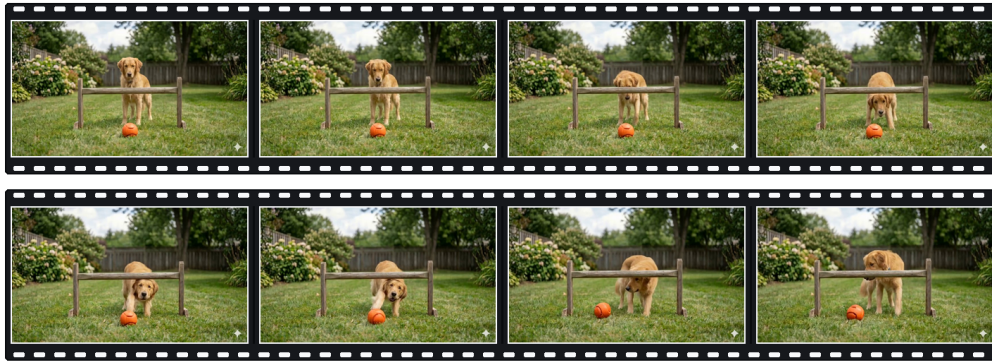


Figure 30: **Example of the scene “Dog Route Choice Toward Toy” generated by Cosmos 3 Super I2V.** The model is instructed to let the dog make one reach attempt toward the toy. The dog lowers its head under the bar, then backs away and remains behind the obstacle.