REWARD AS AN AGENT FOR EMBODIED WORLD MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

RL for world models largely relies on conservative rollouts near the training distribution, limiting exploration and behavioral diversity. In this work, we challenge this conservative paradigm. We argue that the core limitation is not exploration itself, but the lack of reliable verification strategies to support broader exploration. Without reliable verification, policies can exploit proxy rewards through convincing videos that lack evidence of genuine task progress. On the verification side, we introduce **Reward as an Agent**, which centers evaluation on evidence-grounded task verification. It separates observed behavior from fixed task requirements, actively inspects disputed evidence, and revises judgments through tool-assisted reflection and requirement checks. Reward gates preserve supported progress while withholding credit for failure or unverifiable essential content. The framework is extensible: additional reward strategies can be integrated as tools to enrich verification and support further performance gains. On the exploration side, we introduce **Dynamic-Aware Rollout Diversification** through **DynDiff-GRPO**, which allocates stochastic exploration to dynamically salient regions to diversify trajectories. Together, these designs couple broader exploration with evidence-based reward evaluation. Training experiments demonstrate benchmark improvements across three open-source world models, while complementary offline evaluations show that Reward as an Agent improves reward reasonableness and reduces erroneous positive rewards on failure and unobservable clips.

1 INTRODUCTION

World models support learning, planning, and generation by predicting how observations evolve under actions. As learned simulators, they enable policy optimization and interaction learning from large-scale data without explicit physical modeling (Yang et al., 2023; Assran et al., 2025). World-action models (WAMs) further couple visual prediction with action modeling (Cen et al., 2025; Ye et al., 2026; Li et al., 2026a; Kim et al., 2026; Kairos Team et al., 2026), making reliable evaluation of predicted interactions and task outcomes essential for action learning.

Reinforcement learning (RL) offers a natural approach to world-model post-training. Group-based methods such as GRPO (Shao et al., 2024) have achieved substantial gains in language and vision-language models (Guo et al., 2025; Bai et al., 2025), supported by reliable rewards from code execution or rule-based checks. Recent work extends these methods to visual generation (Liu et al., 2025a; Xue et al., 2025; Li et al., 2025a; Wang et al., 2025; Zheng et al., 2026b; Wang et al., 2026), but reliable reward evaluation remains a bottleneck for embodied world models.

Generated videos lack a direct correctness test: plausible appearance can conceal missing interactions, limited task progress, or physical defects. Broader RL exploration can amplify these ambiguities, allowing policies to exploit visual shortcuts that increase proxy rewards without improving behavior. Reliable evaluation therefore requires both evidence of what occurred and explicit criteria for what the task required.

Our core insight is that **exploration and verification must scale in tandem**. Reward as an Agent first records visible behavior without exposing the observer to the task instruction, while separately extracting and freezing task requirements. It then jointly assesses task completion, physical plausibility, and visual quality. To verify its judgments, the agent revisits selected frames and regions,

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

consults CoTracker3 motion evidence and WMReward prediction-surprise feedback, and checks for omitted requirements and contradictions. Tools inform verification and reflection rather than directly contributing to the reward. The framework also supports additional tool adapters, including human-feedback reward models (He et al., 2024; Liu et al., 2025b), reasoning-based evaluators (He et al., 2025; Wang et al., 2026), and world-model physical priors (Yuan et al., 2026), whose feedback can be checked against visual evidence.

Fixed score aggregation and explicit gates preserve useful partial progress while withholding credit for task failure or unverifiable essential content. Execution and tool faults instead produce ineligible results. This makes reward decisions inspectable, while retaining dependence on the underlying models and tools. Complementing verification, DynDiff-GRPO broadens trajectory diversity by allocating stochastic exploration to dynamically salient regions.

Our contributions are threefold:

- **Reward-as-an-Agent Framework:** We combine task-blind observations and fixed task requirements with joint assessment, adaptive inspection, tool-assisted reflection, and scope repair. Explicit gates distinguish supported progress, zero-credit outcomes, and evaluator faults.
- **Dynamics-Aware Rollout:** We allocate stochastic exploration to dynamic regions and account for the policy-dependent mask in both sampling and Gaussian likelihoods.
- **Empirical Evaluation:** Training experiments demonstrate benchmark improvements across embodied world models, while complementary offline analyses assess reward reasonableness and resistance to erroneous rewards on hacking/control clips.

2 Reward as an Agent

2.1 Reward Hacking in Embodied World Model RL

We formulate denoising as a Markov Decision Process (MDP): the state contains the current latent, denoising time, and conditioning inputs; an action is the next latent sampled by the world-model policy. The completed video receives the task reward, while intermediate denoising rewards are zero:

$$\text{MDP} = (\mathcal{S}, \mathcal{A}, P, R), \qquad J(\pi) = \mathbb{E}_\pi\Big[\sum_{t=0}^{T} R(s_t, a_t)\Big] \tag{1}$$

Under this framework, policy improvement is fundamentally shaped by reward quality, since optimization follows the provided reward signal rather than the true objective itself.

Reward Hacking Definition. This creates a critical vulnerability when reward functions are imperfect. Reward hacking occurs when the policy exploits flaws in proxy rewards rather than genuinely accomplishing the intended task (Skalse et al., 2022):

$$J(\pi_{\text{hack}}) > J(\pi_{\text{intended}}), \quad \text{while} \quad \text{TaskEval}(\pi_{\text{hack}}) \ll \text{TaskEval}(\pi_{\text{intended}}) \tag{2}$$

In other words, optimization succeeds mathematically but fails behaviorally. In embodied diffusion systems, this often manifests through exploiting visual shortcuts, shallow semantic alignment, or model artifacts while bypassing authentic physical interaction.

Empirical Observation of Reward Hacking. This problem is particularly severe in embodied world models because embodied tasks require not only visual realism, but also semantic correctness, physical plausibility, and genuine task completion. However, most existing reward systems are inherited from general video generation and lack embodied-specific evaluation capability. To systematically examine this mismatch, we optimize embodied world models in embodied scenarios using DanceGRPO (Xue et al., 2025) with two representative general-purpose reward metrics, VideoAlign (Liu et al., 2025b) and UnifiedReward-Flex (Wang et al., 2026). Despite continuous optimization, we observe that reward values largely fluctuate within a narrow range while downstream generation quality shows little meaningful improvement. To understand this discrepancy, we conduct detailed analysis of model rollouts alongside their corresponding reward signals, and identify reward-hacking cases in which high proxy scores do not reflect task success. Detailed scoring prompts and evaluation criteria are provided in Appendix C.

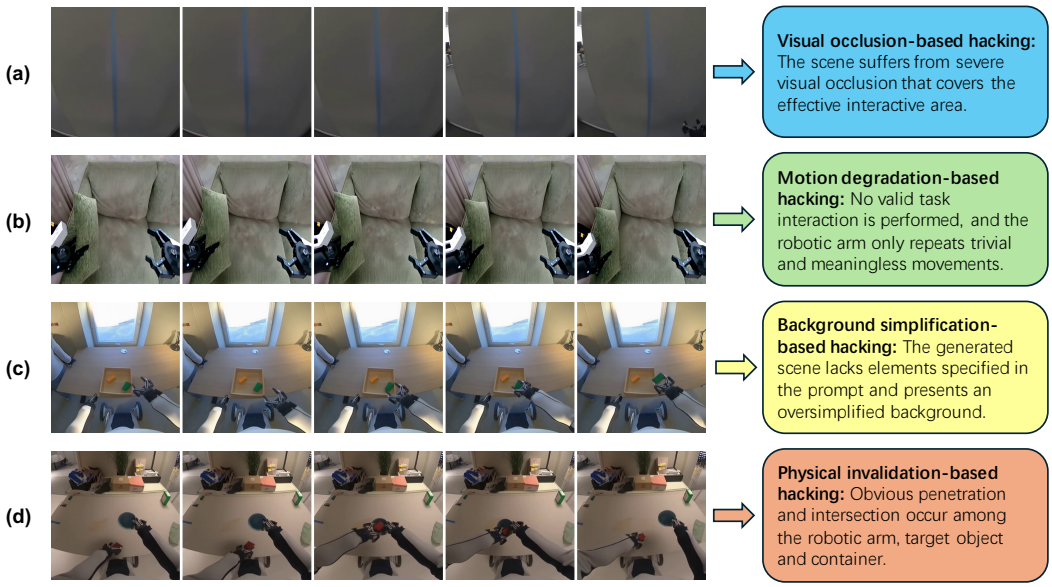


Figure 1: Typical reward hacking failures under existing reward metrics, where generated videos receive deceptively high scores despite violating true task objectives. (a) Visual occlusion: $0.79/1.0$ on VideoAlign VQ; (b) Motion degradation: $2.76/5.0$ on UnifiedReward-Flex Alignment Score; (c) Background simplification: $0.80/1.0$ on VideoAlign VQ; (d) Physical invalidation: $3.0/5.0$ on UnifiedReward-Flex Physics Score. These cases highlight systematic blind spots in current reward designs.

Our analysis reveals four recurring reward hacking patterns (Figure 1): (1) **Visual occlusion**, where blur, shadows, or occlusion conceal physical defects; (2) **Motion degradation**, where the model minimizes risk through static or trivial movement; (3) **Background simplification**, where environmental complexity is reduced to avoid generation failures; and (4) **Physical invalidation**, where outputs superficially satisfy prompts while violating physical laws.

These observations reveal that current reward paradigms fail not because RL optimization itself is ineffective, but because the reward functions guiding optimization are structurally incomplete. Existing systems suffer from four major limitations: (1) perceptual metrics built on traditional computer vision detection models often focus on object recognition, trajectory matching, or pixel-level similarity, and do not directly assess embodied physical interactions; (2) VLM-based evaluators fine-tuned for specific domains may perform well on narrow tasks, but generally exhibit poor domain generalization when transferred beyond their specialized settings; (3) incomplete evaluation dimensions that overlook critical factors such as physical compliance or authentic task completion; and (4) reward aggregation that allows visual or physical scores to compensate for established task failure. As a result, policies are incentivized to maximize proxy scores through exploitable shortcuts rather than genuine embodied competence.

These failures suggest that reward hacking is fundamentally a reward design problem rather than merely an optimization problem: if reward functions remain unreliable, stronger RL algorithms may simply exploit them more effectively. The core challenge is therefore to redesign reward as a robust evaluative system that better aligns proxy objectives with true embodied goals.

2.2 Reward Design

Design through Iterative Evaluation. Our agent aims to verify task progress, reject unsupported credit, and preserve useful partial execution. Inspired by recursive self-improvement (Zhang et al., 2025) and automated agent design (Hu et al., 2024), we iteratively evaluated rewards and explanations on 100 ordinary and 80 hacking/control videos, then revised prompts, tools, and decision rules. This process produced 16 development versions, distinct from the 14 final-design ablations in Sec. 4.2. These development-exposed samples do not establish held-out generalization.

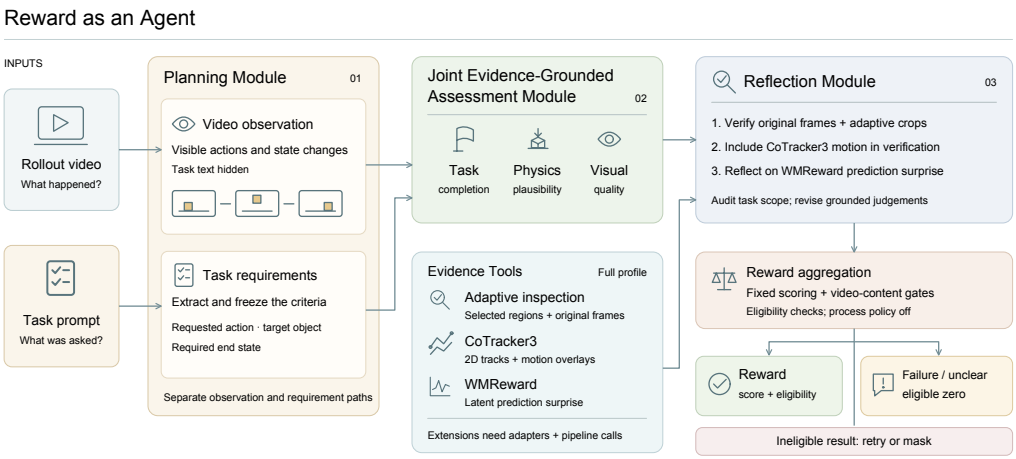


Figure 2: Reward Agent workflow. Separate observations and task requirements support a joint assessment, followed by visual verification, tool-assisted reflection, and requirement repair. Tools provide evidence; final rules determine the reward and training eligibility.

Reward as an Agent combines a multimodal model with CoTracker3 (Karaev et al., 2025) and WMReward (Yuan et al., 2026) (Figure 2). The model gathers evidence and revises a structured assessment; fixed rules convert the final assessment into a reward.

Separate Observation and Task Assessment. The agent first observes 32 uniformly sampled frames without the task instruction. It then extracts and freezes the instruction's required actions, objects, ordering, and outcomes as a *task contract*. Separating these steps reduces the risk of mistaking desired behavior for observed behavior. Given the contract and video evidence, the model assesses *task completion*, *physical plausibility*, and *visual quality*, recording requirement-level judgments and supporting frame references. Useful progress can receive partial credit; mere approach or ineffective contact does not establish success.

Verify and Reflect with Tools. The agent revisits uncertain judgments using original frames, selected crops, and CoTracker3 motion overlays. It reviews all frames of clips with at most 81 frames; longer clips add up to 16 nearby frames to the initial sample. Invalid crop plans fall back to full-frame views. CoTracker3 supplies point trajectories and visibility, while WMReward uses V-JEPA 2 (Assran et al., 2025) to provide latent prediction-surprise feedback for a subsequent visual reflection. Both tools supply supporting evidence: tracks do not certify contact or success, and prediction surprise is neither a physical-validity guarantee nor a direct reward term.

Check Requirements and Resolve Failures. A scope audit checks missing requirements, invented conditions, and contradictions, allowing up to three image-backed repair rounds. Text-only suggestions cannot override visual judgments. When needed, a final visual check determines whether task failure is established despite remaining ambiguity. Required-tool and protocol failures make the evaluation ineligible, rather than triggering model-only fallback. An initial quality check may assign zero to unusable or spatially uniform video content before tool inference.

Extend the Evidence Tools. Additional reward models, task-specific evaluators, or rule-based checks can be integrated through custom adapters and explicit verification or reflection calls. Their feedback must be checked against the task contract and visual evidence before revising the assessment. Our implementation evaluates CoTracker3 and WMReward; it does not provide automatic tool discovery or a generic plug-in registry.

Reward Aggregation and Eligibility. Before target-mismatch adjustment, we map the task, physical, and visual judgments to fixed scores: $T \in \{1, 0.8, 0.45, 0\}$ for complete, mostly complete, partial, and failed execution; $P \in \{1, 0.75, 0.25\}$ for plausible motion, minor defects, and major defects; and $V \in \{1, 0.8, 0.3\}$ for clear, mildly degraded, and severely degraded but assessable imagery. For assessable components, the diagnostic reward is

$$R_{\text{diag}}(v, c) = \big(0.70T + 0.20P + 0.10V\big)\big(0.5 + 0.5P\big). \tag{3}$$

An explicitly mismatched target caps T at 0.2. These mappings and weights are fixed design choices.

Let E indicate that the evaluation passes the required execution and consistency checks, F denote evidence-supported task failure, and U denote video-content uncertainty that prevents justified positive credit. The final reward is

$$R(v,c) = \begin{cases} \perp, & E(v,c) = 0, \\ 0, & E(v,c) = 1 \text{ and } (F(v,c) = 1 \text{ or } U(v,c) = 1), \\ R_{\text{diag}}(v,c), & \text{otherwise.} \end{cases} \tag{4}$$

Supported partial progress retains graded credit, while failure or unverifiable essential content receives zero. Uncertainty-based zero does not imply confirmed failure. Here $\perp$ denotes an invalid evaluation that must be retried or excluded from training; only explicitly eligible results support policy updates. The full Agent uses API-based inference and local tools; runtime settings and an alternative local deployment appear in Appendix E.

3 Dynamic-Aware Rollout Diversification

3.1 Dynamic-Space Under-Exploration in Conservative RL Rollout

Existing RL methods for diffusion world models are often overly conservative, constraining rollouts near the training distribution to avoid reward hacking (Appendix A). While this ensures stability, it severely limits the exploration of diverse dynamics. Crucially, in embodied tasks, policy improvement requires exploring multiple plausible dynamic evolutions within the same scene, even as static elements like environment geometry remain consistent. Standard conservative GRPO fails to distinguish between these dimensions; by uniformly suppressing rollout stochasticity, it collapses diffusion sampling toward deterministic, ODE-like trajectories. This leads to a narrow set of reward-favored behaviors, increasing reward overfitting and stifling the discovery of rich dynamic variety.

Therefore, effective embodied rollout diversification should not simply increase global randomness, but instead preserve scene fidelity while expanding dynamic-space exploration. Motivated by this, we propose **DynDiff-GRPO**, a dynamic-aware stochastic rollout strategy that selectively reallocates stochasticity toward dynamically salient regions while maintaining stable scene consistency.

3.2 DynDiff-GRPO: Dynamic-Aware Stochastic Rollout

Dynamic-Aware Stochastic Rollout Diversification. Given latent state x_t and velocity prediction $v_\theta(x_t, t, c)$, we first estimate the predicted clean sample and noise component:

$$\hat{x}_0 = x_t - \sigma_t v_\theta, \qquad \hat{\epsilon}_t = x_t + (1-\sigma_t) v_\theta. \tag{5}$$

Building on the coefficient-preserving sampling analysis of CPS (Wang & Yu, 2025), which theoretically shows that reducing excessive stochastic perturbation during reverse diffusion can yield a more stable and effective SDE rollout process than conventional flow-SDE sampling, we further generalize this insight into a hybrid ODE/SDE transition framework. Specifically, rather than fixing rollout dynamics, we continuously interpolate between deterministic exploitation and stochastic exploration:

$$x_{t^-} = \hat{x}_0(1-\sigma_{t^-}) + \hat{\epsilon}_t \sqrt{\sigma_{t^-}^2 - (\sigma_{t^-}^{\text{noise}})^2} + \sigma_{t^-}^{\text{noise}} \odot z, \quad z \sim \mathcal{N}(0, I). \tag{6}$$

where $\sigma_{t^-}^{\text{noise}} \in [0, \sigma_{t^-}]$ controls exploration strength, recovering deterministic ODE rollout when $\sigma_{t^-}^{\text{noise}} = 0$ and standard stochastic rollout when $\sigma_{t^-}^{\text{noise}} = \sigma_{t^-}$. Instead of uniformly distributing stochasticity across spatial locations, DynDiff-GRPO reallocates exploration according to dynamic saliency:

For frame index k, we construct the noise scale and dynamic priors as

$$\sigma_{t^-}^{\text{noise}} = \sigma_{t^-} \odot M_t, \qquad R^{(k)} = \Delta\hat{x}_0^{(k)} - \text{Mean}_{h,w}\left(\Delta\hat{x}_0^{(k)}\right), \tag{7}$$

$$D = \text{Normalize}(\|R\|_2), \qquad B = \mathbb{I}[D > \text{Quantile}(D, \tau)]. \tag{8}$$

Here, $\Delta\hat{x}_0^{(k)}$ is the difference between adjacent predicted frames. Subtracting its spatial mean isolates localized dynamics; normalization and percentile thresholding yield the continuous intensity D and sparse prior B.

Finally, continuous dynamic intensity and sparse dynamic-region priors are combined to produce the rollout mask:

$$M_t = r_{\text{base}} + (1 - r_{\text{base}})\,(0.5D + 0.5B)\,, \tag{9}$$

where r_{base} defines the minimum exploration ratio, ensuring that low-dynamic regions preserve a non-zero level of stochasticity while highly dynamic regions receive stronger exploration.

Substituting the adaptive variance into Eq. 6, the rollout transition distribution becomes:

$$x_{t^-} \sim \mathcal{N}\left(\mu, (\sigma_{t^-} \odot M_t)^2\right), \quad (10) \qquad \mu = \hat{x}_0(1 - \sigma_{t^-}) + \hat{\epsilon}_t\sqrt{\sigma_{t^-}^2 - (\sigma_{t^-}^{\text{noise}})^2}. \tag{11}$$

The following expression gives the element-wise log density; summing over latent coordinates yields the transition log-likelihood used during policy optimization:

$$\log p(x_{t^-}) = -\frac{(x_{t^-} - \mu)^2}{2(\sigma_{t^-} \odot M_t)^2} - \log(\sigma_{t^-} \odot M_t) - \log\sqrt{2\pi}. \tag{12}$$

See Sec. 4.3 for quantitative analysis.

Group-Relative Policy Optimization for Diffusion World Models. Using the dynamic-aware stochastic sampler, for each conditioning prompt c, DynDiff-GRPO samples a rollout group $\{\tau_i\}_{i=1}^{G} \sim \pi_{\theta_{\text{old}}}(\cdot|c)$ under independently diversified diffusion paths, producing rewards $r_i = R(\tau_i, c)$. Let $\mathcal{I} = \{i : E(\tau_i, c) = 1\}$ be the eligible rollout indices and $n = |\mathcal{I}|$. Content-based uncertainty can yield an eligible zero; evaluator faults are retried or excluded. Groups with fewer than two eligible rollouts are skipped. For $i \in \mathcal{I}$, we use clipped group-relative advantages:

$$\begin{aligned} \bar{r} = \frac{1}{n}\sum_{j\in\mathcal{I}} r_j, \qquad s_r = \sqrt{\frac{1}{n}\sum_{j\in\mathcal{I}}(r_j - \bar{r})^2}, \\ A_i = \text{clip}\left(\frac{r_i - \bar{r}}{s_r + \epsilon}, -A_{\max}, A_{\max}\right). \end{aligned} \tag{13}$$

Here $\epsilon > 0$ stabilizes normalization; equal-reward groups yield zero advantages. Dynamic awareness enters the rollout mask and policy likelihood, while reward normalization uses the eligible group's standard deviation.

For each denoising step, we model the reverse transition as a Gaussian policy. **Policy-Dependent Variance and Likelihood.** CPS (Wang & Yu, 2025) simplifies its likelihood ratio using a noise scale shared by the current and old policies. In DynDiff-GRPO, the mask is constructed from the model's predicted clean sample, coupling dynamic-aware exploration to the transition distribution used for policy optimization. Evaluating each policy's mask from its own prediction yields the feature-wise standard deviation $s_{\theta,t} = \sigma_{t^-} \odot M_{\theta,t}(x_t, c)$. The Gaussian log-likelihood is

$$\log \pi_\theta(x_{t^-}|x_t, c) = -\sum_j \left[\frac{(x_{t^-,j} - \mu_{\theta,j})^2}{2s_{\theta,t,j}^2} + \log s_{\theta,t,j} + \frac{1}{2}\log(2\pi)\right], \tag{14}$$

where j indexes latent coordinates. This density is defined on stochastic steps $\mathcal{T}_+ = \{t : \sigma_{t^-} > 0\}$, with $0 < r_{\text{base}} \le 1$ ensuring positive coordinate-wise scales. A deterministic terminal transition with $\sigma_{t^-} = 0$ is excluded from likelihood ratios and the policy loss.

For the same sampled transition, the current and old policies can have different dynamic masks and hence different standard-deviation fields. The importance ratio must therefore retain both **variance-weighted residuals and log-scale normalization**. The Gaussian constant cancels at fixed dimensionality, whereas the log-scale contribution $\sum_j \log(s_{\theta_{\text{old}},t,j}/s_{\theta,t,j})$ generally does not. If both policies share a fixed mask, the log-scale terms cancel, but the residuals remain weighted by the spatial variance.

Table 1: Performance on the robotics subset of PAI-Bench. Numbers in parentheses indicate absolute improvements over the corresponding base model. Bold values improve over the respective baseline. All models and their DynDiff-GRPO counterparts use the same Qwen Flash evaluator and evaluation settings.

Model	Param	i2v-bg	i2v-s	aes	img	bg-con	mot	sub-con	o-con	Domain ↑	Overall ↑
Cosmos-Predict2.5	2B	93.47	91.13	47.74	70.45	90.86	98.80	90.06	19.30	83.61	79.42
DynDiff-GRPO (Ours)	2B	**96.48**	**96.97**	**49.14**	**70.60**	90.47	**99.32**	**91.99**	**19.68**	**84.41** (+0.80)	**80.62** (+1.20)
Kairos3.0-Robot	4B	97.75	95.00	46.72	70.78	92.31	99.26	93.32	19.56	83.87	80.35
DynDiff-GRPO (Ours)	4B	97.17	**96.05**	**46.94**	**71.32**	**92.95**	99.25	91.11	**19.68**	**86.95** (+3.08)	**81.88** (+1.53)
Kairos3.1	4B	99.22	95.06	45.40	67.54	94.90	99.47	95.71	19.46	85.43	81.26
DynDiff-GRPO (Ours)	4B	**99.44**	**96.12**	**46.83**	**70.16**	94.87	99.22	95.63	**19.67**	**88.31** (+2.88)	**83.03** (+1.77)

This couples dynamic-aware exploration with GRPO updates that account for changes in both the transition mean and scale. We compute the importance ratio using each policy's corresponding Gaussian likelihood:

$$\rho_{i,t} = \exp\Big(\log \pi_\theta(x_{t^-}^{(i)}|x_t^{(i)}, c) - \log \pi_{\theta_{\text{old}}}(x_{t^-}^{(i)}|x_t^{(i)}, c)\Big)\,. \tag{15}$$

The policy objective follows GRPO (Shao et al., 2024), using the clipped probability-ratio surrogate introduced in PPO (Schulman et al., 2017), with the expectation restricted to $i \in \mathcal{I}$ and $t \in \mathcal{T}_+$:

$$\mathcal{L}_{\text{GRPO}} = -\mathbb{E}_{i,t}[\min(\rho_{i,t}A_i,\ \text{clip}(\rho_{i,t}, 1-\delta, 1+\delta)A_i)]\,. \tag{16}$$

For regularization, following CPS and SAGE-GRPO, we constrain policy updates toward the target distribution with a mean-space KL-style regularizer and combine it with the GRPO objective:

$$\mathcal{L}_{\mu\text{-KL}} = \left\| \frac{\mu_\theta - \mu_{\text{target}}}{\sigma_{\text{target}} + \epsilon} \right\|_2^2. \tag{17}$$

$$\mathcal{L} = \mathcal{L}_{\text{GRPO}} + \beta \mathcal{L}_{\mu\text{-KL}}. \tag{18}$$

Through dynamic-aware rollout diversification, dynamic-aware vectorized Gaussian likelihood estimation, clipped group-relative optimization, and mean-space regularization, DynDiff-GRPO transforms expanded dynamic-space exploration into stable and reward-aligned diffusion policy improvement. See Appendix F for rollout implementation details.

4 Experimental Results

4.1 Main Experiment

Evaluation settings. We evaluate on PAI-Bench's robotics subset (174 tasks, 913 VQA questions) (Zhou et al., 2025). On a 0–100 scale, Domain is question-level accuracy, Quality averages the eight visual metrics in Table 1, and Overall averages both. All models are evaluated using Qwen3.8-Flash-Next with eight uniformly sampled frames, fixed binary prompts, and greedy decoding (temperature 0, top-p 1, thinking disabled); repeated scoring of identical videos under the same configuration reproduces the results.

Experimental setup. We train Cosmos-Predict2.5-2B (NVIDIA et al., 2025) and Kairos-3.0-Robot (Mar. Version) (Kairos Team et al., 2026), with supplementary Kairos3.1 (4B) experiments using the Kairos3.0 configuration. All runs use Doubao 2.1 Pro as the Reward Agent backbone (Sec. 4.2). Training settings are listed in Appendix G, with configuration analyses in Appendix B.

Training data. Public training data are sampled to match PAI-Bench's robot-embodiment proportions, with sequences, initial frames, and prompts disjoint from the test set. Only the initial frame and an augmented task prompt condition each rollout; subsequent source frames are not training targets.

Main Results. Table 1 shows Domain gains of 0.80, 3.08, and 2.88 points on Cosmos, Kairos3.0, and Kairos3.1, with Overall gains of 1.20, 1.53, and 1.77. Kairos3.1 reaches 88.31 Domain and 83.03 Overall; its mean Quality increases by 0.65 to 77.74, with five of eight metrics improving. Gains are computed relative to each model's corresponding baseline under the same evaluation settings.

4.2 Reward Analysis

Setup. We evaluate 14 configurations per backbone (Doubao 2.1 Pro and Qwen3.8-Flash-Next) on 180 videos: 100 ordinary and 80 hacking/control clips. Nine configurations cumulatively extend direct scoring; five remove individual modules from the full agent, preserving unaffected evidence and downstream settings. We also evaluate VideoAlign (Liu et al., 2025b) and UnifiedReward-Flex (Wang et al., 2026) as standalone reward baselines. Direct scoring receives only the video, instruction, and scoring criteria. Identical stage inputs share outputs within each evaluation run.

Reward Reasonableness. Doubao 2.1 Pro (ByteDance Seed Team, 2026) judges both backbones using the video, instruction, reward, and explanation, with method identities hidden, up to 64 frames, and temperature 0. It accepts scores consistent with task progress and execution quality without decisive reasoning errors, while tolerating minor calibration differences. Unverifiable essential content warrants zero; execution errors are retried. We report the accepted fraction of 180 outputs (prompt in Appendix H.4).

Table 2: Reward reasonableness and hacking/control errors. R: reasonable judgments among 180 videos (Doubao 2.1 Pro judge); F+: positive rewards on 60 reference-negative clips; P0: zero rewards on 20 progress clips. Standalone rewards use their own models; Agent columns use Doubao 2.1 Pro and Qwen Flash. Modules are added cumulatively (middle) or removed individually (bottom). Bold: best per backbone, including standalone baselines.

Method	R / 180 (%) ↑		F+ / 60 ↓		P0 / 20 ↓	
VideoAlign (Liu et al., 2025b)	35 (19.4%)		60		**0**	
UnifiedReward-Flex (Wang et al., 2026)	42 (23.3%)		60		**0**	
Configuration	Doubao	Qwen	Doubao	Qwen	Doubao	Qwen
Direct base-model scoring	132 (73.3%)	115 (63.9%)	50	57	1	**0**
+ Structured report and score mapping	140 (77.8%)	140 (77.8%)	47	52	1	3
+ Task decomposition	156 (86.7%)	138 (76.7%)	53	52	1	4
+ Task-blind observation	170 (94.4%)	145 (80.6%)	42	55	2	5
+ Adaptive visual verification	169 (93.9%)	146 (81.1%)	35	42	1	2
+ CoTracker3 motion evidence	170 (94.4%)	148 (82.2%)	37	47	3	3
+ WMReward feedback and reflection	172 (95.6%)	146 (81.1%)	40	52	3	3
+ Scope audit and repair (without failure gate)	174 (96.7%)	152 (84.4%)	48	51	3	1
Full Reward Agent	**178 (98.9%)**	**159 (88.3%)**	**0**	3	**0**	1
Without task-blind observation	169 (93.9%)	146 (81.1%)	6	6	2	**0**
Without adaptive visual verification	169 (93.9%)	138 (76.7%)	5	3	4	4
Without WMReward feedback and reflection	171 (95.0%)	142 (78.9%)	5	4	4	4
Without scope audit and repair	174 (96.7%)	141 (78.3%)	5	**2**	4	5
Without CoTracker3 motion evidence	170 (94.4%)	146 (81.1%)	6	3	4	3

Results. The full agent achieves the highest reasonableness: 98.9% for Doubao and 88.3% for Qwen Flash, exceeding direct scoring by 25.6 and 24.4 percentage points. Individual module removals, including the failure gate, reduce reasonable judgments by 4–9 and 7–21, respectively, indicating complementary benefits. VideoAlign and UnifiedReward-Flex achieve 19.4% and 23.3% reasonableness; both assign positive rewards to all 60 reference-negative clips, despite producing no false zeros on the 20 progress clips.

Hacking/control evaluation. The 80 clips comprise 57 failures, 3 unobservable cases, and 20 with partial or complete progress, according to automatic reference labels withheld from the judge. These labels are not human ground truth. Table 2 reports positive rewards on the first 60 clips (F+) and zero rewards on the latter 20 (P0).

The failure gate drives the largest reduction in erroneous positive rewards: with all other modules fixed, F+ falls from 48/60 to 0/60 for Doubao and from 51/60 to 3/60 for Qwen. P0 also improves from 3/20 to 0/20 for Doubao and remains 1/20 for Qwen, showing that this reduction does not sacrifice supported progress. Other modules help balance rejection and progress: removing scope audit lowers Qwen's F+ from 3 to 2 but raises P0 from 1 to 5 and lowers reasonableness to 78.3%. The Agent supports both the closed-source Doubao 2.1 Pro and the open-source Qwen Flash backbone.

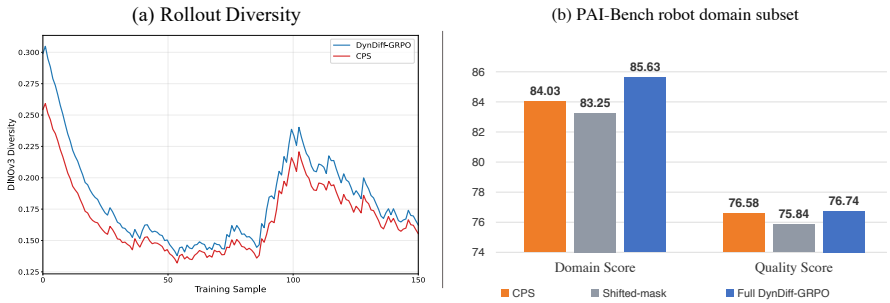


Figure 3: Ablation of rollout strategies. (a) Rollout diversity via mean pairwise DINOv3 cosine distance. (b) Domain and Quality scores on the PAI-Bench robot domain subset for CPS, shifted-mask DynDiff-GRPO, and full DynDiff-GRPO. Quality averages eight visual metrics; higher is better.

We select Doubao 2.1 Pro for the final reward system and the training experiments in Table 1, based on its higher reasonableness and zero errors on both hacking/control metrics in this evaluation.

4.3 Rollout Strategy Analysis

We compare DynDiff-GRPO with CPS (Wang & Yu, 2025) through rollout diversity and PAI-Bench ablations. **Rollout Diversity Analysis.** For each video, we concatenate L_2-normalized DINOv3 (Siméoni et al., 2025) features from T uniformly sampled frames, then normalize the concatenation to obtain $\hat{z}_v$. Diversity across N rollouts per prompt is their mean pairwise cosine distance:

$$D_{\text{prompt}} = \frac{2}{N(N-1)} \sum_{1 \le i < j \le N} \left(1 - \hat{z}_i^\top \hat{z}_j\right). \tag{19}$$

Figure 3(a) shows that DynDiff-GRPO consistently maintains higher diversity than CPS during training, indicating greater diversity in this feature space; this metric alone does not establish physical validity.

Robotics Evaluation on PAI-Bench. All variants use Kairos-3.0-Robot (Kairos Team et al., 2026), identical training data, and 480×832 resolution. Reward Agent, training steps, rollout counts, CFG, and KL regularization are fixed; only the rollout strategy varies. Figure 3(b) compares CPS, shifted-mask DynDiff-GRPO, and full DynDiff-GRPO. Relative to CPS, the full method improves Domain by 1.60 points (85.63 vs. 84.03) while maintaining similar Quality (76.74 vs. 76.58).

Spatial-Alignment Ablation. We circularly shift each dynamic mask by half the latent height and width, preserving its value distribution but displacing exploration. This shift is fixed across frames and denoising steps in both sampling and likelihood evaluation. In Fig. 3(b), full DynDiff-GRPO exceeds this control by 2.38 Domain points (85.63 vs. 83.25) and 0.90 Quality points (76.74 vs. 75.84). The control also underperforms CPS, supporting the benefit of aligning exploration with dynamic regions.

5 Conclusion and Limitations

We present Reward as an Agent for evidence-based task verification and DynDiff-GRPO for exploration in dynamically salient regions. Experiments show benchmark gains across three world models; complementary offline evaluations show more reasonable rewards and fewer hacking/control errors.

Limitations. Visual and tool-based evidence can misinterpret interactions or occluded outcomes. Automatic judging on development-exposed data may overestimate generalization, particularly with a same-family judge. Independent human evaluation, matched online RL trials, and serving-cost measurements remain necessary; video benchmarks alone do not establish closed-loop robot-control gains.

AI USE STATEMENT

We used generative AI tools to assist with manuscript drafting and editing, translation, clarification of mathematical notation and formulations, code review, interpretation of experimental results, and consistency checks across text, tables, and references. AI tools also assisted with LaTeX formatting and figure preparation. Separately, the language models used within our reward and evaluation pipelines are described in the methods and experimental sections. The authors take responsibility for the final content of this paper, including all AI-assisted text, claims, and artifacts.

REFERENCES

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

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

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mahmoud Assran, and Nicolas Ballas. Revisiting Feature Prediction for Learning Visual Representations from Video. arXiv preprint arXiv:2404.08471, 2024. URL https://arxiv.org/abs/2404.08471.

Jake Bruce, Michael Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, Yusuf Aytar, Sarah Bechtle, Feryal Behbahani, Stephanie Chan, Nicolas Heess, Lucy Gonzalez, Simon Osindero, Sherjil Ozair, Scott Reed, Jingwei Zhang, Konrad Zolna, Jeff Clune, Nando de Freitas, Satinder Singh, and Tim Rocktäschel. Genie: Generative Interactive Environments. arXiv preprint arXiv:2402.15391, 2024. URL https://arxiv.org/abs/2402.15391.

ByteDance Seed Team. Seed2.1 Officially Released: Advancing AI Productivity. Official ByteDance Seed model release, June 2026. URL https://seed.bytedance.com/en/blog/seed2-1-officially-released-advancing-ai-productivity.

Jun Cen, Chaohui Yu, Hangjie Yuan, Yuming Jiang, Siteng Huang, Jiayan Guo, Xin Li, Yibing Song, Hao Luo, Fan Wang, Deli Zhao, and Hao Chen. WorldVLA: Towards Autoregressive Action World Model. arXiv preprint arXiv:2506.21539, 2025. URL https://arxiv.org/abs/2506.21539.

Patrick Esser, Sumith Kulal, Andreas Blattmann, et al. Scaling Rectified Flow Transformers for High-Resolution Image Synthesis. arXiv preprint arXiv:2403.03206, 2024. URL https://arxiv.org/abs/2403.03206.

Zhen Fang, Wenxuan Huang, Yu Zeng, Yiming Zhao, Shuang Chen, Kaituo Feng, Yunlong Lin, Lin Chen, Zehui Chen, Shaosheng Cao, and Feng Zhao. Flow-OPD: On-policy distillation for flow matching models. https://arxiv.org/abs/2605.08063, 2026. URL https://arxiv.org/abs/2605.08063.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, September 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL http://dx.doi.org/10.1038/s41586-025-09422-z.

Xuan He, Dongfu Jiang, Ge Zhang, Max Ku, Achint Soni, Sherman Siu, Haonan Chen, Abhranil Chandra, Ziyan Jiang, Aaran Arulraj, Kai Wang, Quy Duc Do, Yuansheng Ni, Bohan Lyu, Yaswanth Narsupalli, Rongqi Fan, Zhiheng Lyu, Yuchen Lin, and Wenhu Chen. VideoScore: Building Automatic Metrics to Simulate Fine-grained Human Feedback for Video Generation. arXiv preprint arXiv:2406.15252, 2024. URL https://arxiv.org/abs/2406.15252.

Xuan He, Dongfu Jiang, Ping Nie, Minghao Liu, Zhengxuan Jiang, Mingyi Su, Wentao Ma, Junru Lin, Chun Ye, Yi Lu, Keming Wu, Benjamin Schneider, Quy Duc Do, Zhuofeng Li, Yiming Jia, Yuxuan Zhang, Guo Cheng, Haozhe Wang, Wangchunshu Zhou, Qunshu Lin, Yuanxing Zhang, Ge Zhang, Wenhao Huang, and Wenhu Chen. VideoScore2: Think before You Score in Generative Video Evaluation. arXiv preprint arXiv:2509.22799, 2025. URL https://arxiv.org/abs/2509.22799.

Jonathan Ho and Tim Salimans. Classifier-Free Diffusion Guidance. arXiv preprint arXiv:2207.12598, 2022. URL https://arxiv.org/abs/2207.12598.

Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. GAIA-1: A Generative World Model for Autonomous Driving. arXiv preprint arXiv:2309.17080, 2023. URL https://arxiv.org/abs/2309.17080.

Shengran Hu, Cong Lu, and Jeff Clune. Automated Design of Agentic Systems. arXiv preprint arXiv:2408.08435, 2024. URL https://arxiv.org/abs/2408.08435.

Kairos Team, Fei Wang, Shan You, Qiming Zhang, Tao Huang, Zuoyi Fu, Zhisheng Zheng, Yunlong Xi, Feng Lv, Xiaoming Wu, Zeyu Liu, Cong Wan, Pu Li, Ruiqing Yang, Xiaoou Li, Wei Wang, Kangkang Zhu, Yuwei Zhang, Shi Fu, Zheng Zhang, Xiaoning Wu, Xuzeng Fan, Dacheng Tao, and Xiaogang Wang. Kairos: A Regret-Aware Native World-Action Model Stack for Physical AI. arXiv preprint arXiv:2606.16533, 2026. URL https://arxiv.org/abs/2606.16533.

Nikita Karaev, Yuri Makarov, Jianyuan Wang, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. CoTracker3: Simpler and Better Point Tracking by Pseudo-Labelling Real Videos.

In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6013–6022, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Karaev_CoTracker3_Simpler_and_Better_Point_Tracking_by_Pseudo-Labelling_Real_Videos_ICCV_2025_paper.html.

Moo Jin Kim, Yihuai Gao, Tsung-Yi Lin, Yen-Chen Lin, Yunhao Ge, Grace Lam, Percy Liang, Shuran Song, Ming-Yu Liu, Chelsea Finn, and Jinwei Gu. Cosmos Policy: Fine-Tuning Video Models for Visuomotor Control and Planning. arXiv preprint arXiv:2601.16163, 2026. URL https://arxiv.org/abs/2601.16163.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient Memory Management for Large Language Model Serving with PagedAttention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Junzhe Li, Yutao Cui, Tao Huang, Weijie Kong, Yiming Cheng, Chuxuan Zeng, Yinping Ma, Chun Fan, Miles Yang, Zhao Zhong, and Liefeng Bo. MixGRPO: Unlocking Flow-based GRPO Efficiency with Mixed ODE-SDE. arXiv preprint arXiv:2507.21802, 2025a. URL https://arxiv.org/abs/2507.21802.

Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Fei Han, Mingrui Yu, Zelin Gao, Nan Xue, Xing Zhu, Yujun Shen, and Yinghao Xu. Causal World Modeling for Robot Control. arXiv preprint arXiv:2601.21998, 2026a. URL https://arxiv.org/abs/2601.21998.

Quanhao Li, Junqiu Yu, Kaixun Jiang, Yujie Wei, Zhen Xing, Pandeng Li, Ruihang Chu, Shiwei Zhang, Yu Liu, and Zuxuan Wu. DiffusionOPD: A unified perspective of on-policy distillation in diffusion models. https://arxiv.org/abs/2605.15055, 2026b. URL https://arxiv.org/abs/2605.15055.

Zongjian Li, Zheyuan Liu, Qihui Zhang, Bin Lin, Feize Wu, Shenghai Yuan, Zhiyuan Yan, Yang Ye, Wangbo Yu, Yuwei Niu, Shaodong Wang, Xinhua Cheng, and Li Yuan. Uniworld-V2: Reinforce image editing with diffusion negative-aware finetuning and MLLM implicit feedback. https://arxiv.org/abs/2510.16888, 2025b. URL https://arxiv.org/abs/2510.16888.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-GRPO: Training Flow Matching Models via Online RL. arXiv preprint arXiv:2505.05470, 2025a. URL https://arxiv.org/abs/2505.05470.

Jie Liu, Gongye Liu, Jiajun Liang, Ziyang Yuan, Xiaokun Liu, Mingwu Zheng, Xiele Wu, Qiulin Wang, Menghan Xia, Xintao Wang, Xiaohong Liu, Fei Yang, Pengfei Wan, Di Zhang, Kun Gai, Yujiu Yang, and Wanli Ouyang. Improving Video Generation with Human Feedback. arXiv preprint arXiv:2501.13918, 2025b. URL https://arxiv.org/abs/2501.13918.

Jie Liu, Zilyu Ye, Linxiao Yuan, Shenhan Zhu, Yu Gao, Jie Wu, Kunchang Li, Xionghui Wang, Xiaonan Nie, Weilin Huang, and Wanli Ouyang. UniGRPO: Unified Policy Optimization for Reasoning-Driven Visual Generation. arXiv preprint arXiv:2603.23500, 2026. URL https://arxiv.org/abs/2603.23500.

Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*, 2019. URL https://arxiv.org/abs/1711.05101.

NVIDIA, Arslan Ali, Junjie Bai, Maciej Bala, Yogesh Balaji, Aaron Blakeman, Tiffany Cai, Jiaxin Cao, Tianshi Cao, Elizabeth Cha, Yu-Wei Chao, Prithvijit Chattopadhyay, Mike Chen, Yongxin Chen, Yu Chen, Shuai Cheng, Yin Cui, Jenna Diamond, Yifan Ding, Jiaojiao Fan, Linxi Fan, Liang Feng, Francesco Ferroni, Sanja Fidler, Xiao Fu, Ruiyuan Gao, Yunhao Ge, Jinwei Gu, Aryaman Gupta, Siddharth Gururani, Imad El Hanafi, Ali Hassani, Zekun Hao, Jacob Huffman, Joel Jang, Pooya Jannaty, Jan Kautz, Grace Lam, Xuan Li, Zhaoshuo Li, Maosheng Liao, Chen-Hsuan Lin, Tsung-Yi Lin, Yen-Chen Lin, Huan Ling, Ming-Yu Liu, Xian Liu, Yifan Lu, Alice Luo, Qianli Ma, Hanzi Mao, Kaichun Mo, Seungjun Nah, Yashraj Narang, Abhijeet Panaskar, Lindsey Pavao, Trung Pham, Morteza Ramezanali, Fitsum Reda, Scott Reed, Xuanchi Ren, Haonan Shao, Yue

Shen, Stella Shi, Shuran Song, Bartosz Stefaniak, Shangkun Sun, Shitao Tang, Sameena Tasmeen, Lyne Tchapmi, Wei-Cheng Tseng, Jibin Varghese, Andrew Z. Wang, Hao Wang, Haoxiang Wang, Heng Wang, Ting-Chun Wang, Fangyin Wei, Jiashu Xu, Dinghao Yang, Xiaodong Yang, Haotian Ye, Seonghyeon Ye, Xiaohui Zeng, Jing Zhang, Qinsheng Zhang, Kaiwen Zheng, Andrew Zhu, and Yuke Zhu. World Simulation with Video Foundation Models for Physical AI. arXiv preprint arXiv:2511.00062, 2025. URL https://arxiv.org/abs/2511.00062.

William Peebles and Saining Xie. Scalable Diffusion Models with Transformers. arXiv preprint arXiv:2212.09748, 2022. URL https://arxiv.org/abs/2212.09748.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal Policy Optimization Algorithms. arXiv preprint arXiv:1707.06347, 2017. URL https://arxiv.org/abs/1707.06347.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. arXiv preprint arXiv:2402.03300, 2024. URL https://arxiv.org/abs/2402.03300.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3. arXiv preprint arXiv:2508.10104, 2025. URL https://arxiv.org/abs/2508.10104.

Joar Skalse, Nikolaus H. R. Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and Characterizing Reward Hacking. arXiv preprint arXiv:2209.13085, 2022. URL https://arxiv.org/abs/2209.13085.

Feng Wang and Zihao Yu. Coefficients-Preserving Sampling for Reinforcement Learning with Flow Matching. arXiv preprint arXiv:2509.05952, 2025. URL https://arxiv.org/abs/2509.05952.

Yibin Wang, Zhimin Li, Yuhang Zang, Yujie Zhou, Jiazi Bu, Chunyu Wang, Qinglin Lu, Cheng Jin, and Jiaqi Wang. Pref-GRPO: Pairwise Preference Reward-based GRPO for Stable Text-to-Image Reinforcement Learning. arXiv preprint arXiv:2508.20751, 2025. URL https://arxiv.org/abs/2508.20751.

Yibin Wang, Yuhang Zang, Feng Han, Jiazi Bu, Yujie Zhou, Cheng Jin, and Jiaqi Wang. Unified Personalized Reward Model for Vision Generation. *arXiv preprint arXiv:2602.02380*, 2026.

Zeyue Xue, Jie Wu, Yu Gao, Fangyuan Kong, Lingting Zhu, Mengzhao Chen, Zhiheng Liu, Wei Liu, Qiushan Guo, Weilin Huang, and Ping Luo. DanceGRPO: Unleashing GRPO on Visual Generation. arXiv preprint arXiv:2505.07818, 2025. URL https://arxiv.org/abs/2505.07818.

Sherry Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Leslie Kaelbling, Dale Schuurmans, and Pieter Abbeel. Learning Interactive Real-World Simulators. arXiv preprint arXiv:2310.06114, 2023. URL https://arxiv.org/abs/2310.06114.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, Ayaan Malik, Kyungmin Lee, William Liang, Nadun Ranawaka, Jiasheng Gu, Yinzhen Xu, Guanzhi Wang, Fengyuan Hu, Avnish Narayan, Johan Bjorck, Jing Wang, Gwanghyun Kim, Dantong Niu, Ruijie Zheng, Yuqi Xie, Jimmy Wu, Qi Wang, Ryan Julian, Danfei Xu, Yilun Du, Yevgen Chebotar, Scott Reed, Jan Kautz, Yuke Zhu, Linxi Jim Fan, and Joel Jang. World Action Models are Zero-shot Policies. arXiv preprint arXiv:2602.15922, 2026. URL https://arxiv.org/abs/2602.15922.

Jianhao Yuan, Xiaofeng Zhang, Felix Friedrich, Nicolas Beltran-Velez, Melissa Hall, Reyhane Askari-Hemmat, Xiaochuang Han, Nicolas Ballas, Michal Drozdzal, and Adriana Romero-Soriano. Inference-time Physics Alignment of Video Generative Models with Latent World Models. arXiv preprint arXiv:2601.10553, 2026. URL https://arxiv.org/abs/2601.10553.

Jenny Zhang, Shengran Hu, Cong Lu, Robert Lange, and Jeff Clune. Darwin Godel Machine: Open-Ended Evolution of Self-Improving Agents. arXiv preprint arXiv:2505.22954, 2025. URL https://arxiv.org/abs/2505.22954.

Kaiwen Zheng, Huayu Chen, Haotian Ye, Haoxiang Wang, Qinsheng Zhang, Kai Jiang, Hang Su, Stefano Ermon, Jun Zhu, and Ming-Yu Liu. DiffusionNFT: Online diffusion reinforcement with forward process. In *International Conference on Learning Representations*, 2026a. URL https://arxiv.org/abs/2509.16117.

Mingzhe Zheng, Weijie Kong, Yue Wu, Dengyang Jiang, Yue Ma, Xuanhua He, Bin Lin, Kaixiong Gong, Zhao Zhong, Liefeng Bo, Qifeng Chen, and Harry Yang. Manifold-Aware Exploration for Reinforcement Learning in Video Generation. *arXiv preprint arXiv:2603.21872*, 2026b.

Fengzhe Zhou, Jiannan Huang, Jialuo Li, Deva Ramanan, and Humphrey Shi. PAI-Bench: A Comprehensive Benchmark For Physical AI. arXiv preprint arXiv:2512.01989, 2025. URL https://arxiv.org/abs/2512.01989.

A Related Work

Embodied Video World Models. Recent progress has shifted video generation from passive visual synthesis toward **active** embodied world modeling, where generative and predictive models serve as interactive simulators of physical environments. Early systems such as DeepMind's Genie (Bruce et al., 2024) demonstrated the potential for learning controllable virtual worlds from large-scale unlabeled video. Related real-world simulators include GAIA-1 (Hu et al., 2023) and UniSim (Yang et al., 2023), focusing on autonomous driving and robotic manipulation. NVIDIA Cosmos (NVIDIA et al., 2025) develops video foundation models for physical AI, while V-JEPA (Bardes et al., 2024) learns predictive visual representations and V-JEPA 2 (Assran et al., 2025) extends these toward prediction and robotic planning. Complementary rectified-flow work (Esser et al., 2024) establishes scalable high-resolution image synthesis; it does not by itself demonstrate embodied physical compliance. By unifying action-conditioned prediction with scalable generative stability, these models are evolving into foundation world models that bridge the gap between high-fidelity simulation and downstream policy learning. Building on this connection, world-action models further integrate visual dynamics prediction with robot action generation. WorldVLA unifies autoregressive image and action modeling (Cen et al., 2025), while DreamZero jointly predicts future video and actions using a pretrained video diffusion backbone (Ye et al., 2026). LingBot-VA learns frame prediction and policy execution within an autoregressive diffusion framework (Li et al., 2026a). Cosmos Policy adapts a video model to generate actions, future states, and values for control and planning (Kim et al., 2026). These approaches motivate our focus on evaluating the task evidence in predicted videos, so that reward-guided improvements target meaningful behavior.

Reward Models for Visual Generation. Reward-model design spans supervised quality prediction, preference learning, reasoning-based assessment, and model-derived physical signals. VideoScore (He et al., 2024) trains a multimodal evaluator on human annotations covering multiple aspects of video quality. VideoReward, released through VideoAlign (Liu et al., 2025b), instead learns from multidimensional pairwise preferences over visual quality, motion quality, and text alignment. These designs provide learned scoring signals for evaluating or optimizing generated videos.

Reasoning-based models make the assessment process more explicit. VideoScore2 (He et al., 2025) produces rationales alongside scores for visual quality, text alignment, and physical/common-sense consistency, using supervised fine-tuning followed by GRPO. UnifiedReward-Flex (Wang et al., 2026) constructs context-dependent evaluation criteria under predefined and self-generated dimensions, combining distilled reasoning supervision with preference optimization. A complementary approach, WMReward (Yuan et al., 2026), uses a latent world model's physical prior to guide inference-time video generation. Thus, existing methods offer different forms of feedback, including scores, explanations, adaptive criteria, and physical evidence.

Our framework organizes such feedback within an extensible verification process. A reward model's score or rationale can be exposed through a tool adapter and checked against the instruction and video before influencing the final judgment. Our implementation uses WMReward together with motion tracking; the other reward models discussed here are potential extensions, not additional evaluated components. Their integration could broaden evaluation coverage, while explicit task requirements, evidence checks, and final gates remain responsible for determining whether positive credit is justified.

Reinforcement Learning for World Model Post-Training. Recent post-training methods for generative world models increasingly adopt GRPO (Shao et al., 2024) as a scalable alternative to Actor-Critic RL, enabling policy optimization without auxiliary value networks. Notably, Flow-GRPO (Liu et al., 2025a) pioneered online RL in flow matching by converting deterministic ODE rollouts into stochastic SDE trajectories, establishing a critical balance between exploration and controllability. To address the noise artifacts induced by SDE-based stochasticity, Flow-CPS (Wang & Yu, 2025) reformulates the sampling process to maintain the coefficient-preserving property, thereby enabling more accurate reward modeling and stable convergence. Subsequent variants have largely refined this stability by enforcing rigorous distributional and sampling constraints. DanceGRPO (Xue et al., 2025) extends GRPO across visual generation tasks. UniGRPO (Liu et al., 2026) jointly optimizes reasoning and image generation, replacing latent KL regularization with a velocity-field MSE penalty. MixGRPO (Li et al., 2025a) confines stochastic sampling and optimization to a sliding denoising window to improve efficiency, whereas Pref-GRPO (Wang et al., 2025) uses pairwise

preference rewards to mitigate pointwise reward hacking. Furthermore, frameworks such as SAGE-GRPO (Zheng et al., 2026b) incorporate trust-region restrictions to stabilize policy updates, while systems built on Cosmos-Predict (NVIDIA et al., 2025) often leverage temporal consistency priors to ensure physical plausibility. Beyond reverse-process policy gradients, DiffusionNFT (Zheng et al., 2026a) performs likelihood-free reinforcement through forward-process flow matching, using reward-weighted positive and negative objectives while allowing arbitrary sampling solvers. UniWorld-V2 (Li et al., 2025b) extends this approach to image editing with multimodal-model feedback and group filtering to reduce scoring noise. These advances improve optimization efficiency and stability, but do not by themselves resolve reward hacking or ensure sampling diversity in embodied world models: policy updates still depend on the supplied reward, and solver flexibility does not guarantee diverse task-relevant interaction trajectories. This leaves reliable task verification and explicit rollout diversification as complementary needs, motivating our combination of Reward as an Agent and DynDiff-GRPO.

Recent on-policy distillation (OPD) methods address multi-reward alignment by consolidating specialized teachers. DiffusionOPD (Li et al., 2026b) trains task-specific teachers separately and distills them along student-generated trajectories; Flow-OPD (Fang et al., 2026) further incorporates task routing and manifold anchoring. Although these designs mitigate cross-task interference, training and maintaining multiple teachers followed by a separate distillation stage complicates the training pipeline. Their objectives primarily integrate teacher capabilities rather than explicitly model correlations and dependencies among reward dimensions for the same generated video. Our agent instead jointly assesses task, physical, and visual evidence, with reward gates preventing favorable auxiliary scores from compensating for task failure.

B Analysis of Key Training Configurations

In addition to algorithmic design, we further investigate several configuration choices that have a substantial impact on training stability and final performance, including the KL regularization strength, advantage clipping strategy, input resolution, and classifier-free guidance (CFG) settings.

KL Regularization Formulations. The full Gaussian log-likelihood in Eq. 14 is used for the GRPO importance ratio. Separately, Eq. 17 supplies a variance-normalized mean-distance regularizer. When the current and target masks yield different variances, this mean-only penalty is a surrogate rather than the full Gaussian KL, which also contains variance-ratio and log-scale terms. Using the surrogate for regularization does not remove those terms from the policy likelihood. Building upon this mean-space surrogate, we further introduce a memory- and compute-efficient implementation for periodic regularization. Conventional RLHF and GRPO frameworks typically maintain a frozen reference model in memory to compute the KL penalty $\mathcal{D}_{KL}(\pi_\theta \| \pi_{\text{ref}})$. For large-scale Diffusion Transformers (DiTs) (Peebles & Xie, 2022), this severely bottlenecks GPU memory and requires an additional computational forward pass during every optimization step.

To address this, we completely eliminate the explicit reference model by repurposing the data buffer's lifecycle as a periodic anchor. During the rollout phase, the active policy weights θ_{old} generate the trajectories. We cache the distribution parameters, specifically the mean $\mu_{\theta_{\text{old}}}$ and standard deviation $\sigma_{\theta_{\text{old}}}$, directly into the rollout buffer. During the subsequent optimization phase, these cached values serve as the reference targets for the μ-KL objective over a predefined number of update steps (N = kl_period_steps). Specifically, we instantiate the targets in Eq. 17 as $\mu_{\text{target}} = \mu_{\theta_{\text{old}}}$ and $\sigma_{\text{target}} = \sigma_{\theta_{\text{old}}}$, yielding our buffer-anchored periodic KL penalty:

$$\mathcal{L}_{\mu\text{-KL}} = \mathbb{E}_{s\sim\mathcal{D}}\left[\left\|\frac{\mu_\theta(s) - \mu_{\theta_{\text{old}}}(s)}{\sigma_{\theta_{\text{old}}}(s) + \epsilon}\right\|_2^2\right]. \tag{20}$$

This buffer-based anchoring strategy balances plasticity and stability while providing two critical systemic advantages:

- **Reduced Reference-Model Memory:** By discarding the explicit frozen reference model, we entirely eliminate the GPU memory cost of hosting a second DiT network. This substantial memory saving allows for larger rollout batch sizes, higher input resolutions, or longer frame generation contexts under the same hardware constraints.
- **Enhanced Training Throughput:** Utilizing cached parameters bypasses the redundant forward pass that would otherwise be required by a reference model during the gradient update step. The periodic updating of the anchor (θ_{old}) naturally restricts policy drift within each buffer lifecycle, achieving the stabilizing effects of a periodically updated reference model but at no additional reference-model forward pass.

Advantage Clipping. Advantage clipping limits extreme group-relative updates. We use a threshold of 2.5 in the reported training configurations (Appendix G). Our reward design uses fixed categorical mappings and evidence gating; it does not use voting or curriculum-based score aggregation. Isolating the effect of the reward distribution on training stability would require a matched reward-design ablation.

Classifier-Free Guidance (CFG). We further study the role of CFG (Ho & Salimans, 2022) during rollout generation in diffusion transformer (DiT) training. As shown in Table 3, enabling CFG improves most quality metrics, leading to slightly higher domain and overall scores while maintaining comparable perceptual quality. This is consistent with prior diffusion modeling results, where CFG strengthens conditional guidance by amplifying conditional signals during sampling. However, these gains come with additional sampling overhead due to extra forward evaluations, which reduces rollout throughput in large-scale RL training. In contrast, disabling CFG removes these extra forward evaluations while still retaining competitive final performance under well-shaped reward signals and proper KL regularization. Overall, our results suggest that CFG is an effective auxiliary enhancement rather than a strictly necessary component. In practice, we often disable CFG under high-resolution training settings (e.g., 640×820 and above) to prioritize rollout efficiency while maintaining strong quantitative and perceptual performance.

Table 3: Ablation study on Kairos3.0-Robot on the PAIbench robot domain subset. We compare DynDiff-GRPO with and without CFG under identical base model, data, and 480×832 resolution. CFG improves domain and overall performance while maintaining strong perceptual quality. Best scores are shown in **bold**.

Method	Quality score								Domain score	Overall score
	i2v-bg	i2v-s	aes	img	bg-con	mot	sub-con	o-con		
w/o CFG	97.13	94.26	**46.55**	**69.55**	91.84	99.27	92.58	**19.54**	85.07	80.71
w/ CFG	**97.89**	**95.03**	46.47	68.98	**92.93**	**99.36**	**93.74**	19.50	**85.63**	**81.18**

Table 4: Ablation study on Kairos3.0-Robot on the PAIbench robot domain subset. We report average quality score, domain score, and overall score under different training/inference resolutions and CFG settings. Best results are shown in **bold**.

Method	Train	Infer	CFG	Quality (Avg.)	Domain	Overall
Kairos-LR	480×832	480×832	Yes	76.74	85.63	81.18
Kairos-HR	640×820	640×820	No	**77.24**	85.6	81.42
Kairos-HR	640×820	720×960	No	76.81	**86.95**	**81.88**

Resolution Selection. We further study the role of resolution selection in RL-based training. In our RL setup, rollout resolution can be selected without requiring a matching-resolution target video. We therefore evaluate both training-resolution changes and inference-time scaling.

As shown in Table 4, we first evaluate a low-resolution setting (480×832) with CFG enabled and observe stable performance across both quality and domain scores. We then increase the training resolution to 640×820 and further test inference-time scaling. Without additional training, raising the inference resolution to 720×960 improves domain alignment and overall performance, although average Quality Score decreases from 77.24 to 76.81. These results suggest that RL-trained models can better exploit higher-resolution inference distributions even when such data is scarce during training, highlighting the benefit of resolution-flexible optimization for improving domain alignment and overall performance.

Overall, these findings suggest that the effectiveness of RL for embodied world models depends not only on algorithmic design, but also on optimization stability (KL and clipping), distribution alignment (resolution and rollout configuration), and exploration control (CFG). To improve reproducibility, Appendix G summarizes the final training settings, which correspond to the empirically stable configurations consistently used across both Kairos3.0-Robot and Cosmos-Predict2.5 experiments.

C Appendix: Detailed Experimental Cases of Reward Hacking in Current Mainstream Reward Systems

This appendix presents the detailed experimental contents of the four reward hacking cases summarized in the main text, including the prompts for embodied video generation in each scenario and the corresponding scores of different reward types.

case (a)

```
prompt:
[robot_arm_brand:eve_robot][robot_arm_type:Dual-Arm][camera_position:
head][scene_bg:office][position:in front of a desk][object:red can,
silver can][task:grasped, grasped] Caption:This video is recorded
using a eve_robot robotic arm (Dual-Arm) equipped with a normal camera
 at the head view.  The robot, equipped with dual arms, begins by
simultaneously grasping two small cylindrical metal cans - a red can
and a silver can - both initially positioned near the center of a red
tray. The left arm moves to grasp the red can, fully closing its
gripper for a stable hold, then lifts it slightly and relocates it to
the left side of the tray. Concurrently, the right arm grasps the
silver can, also using a fully closed gripper, lifts it slightly, and
moves it to the right side of the tray while the left arm holds the
red can. The two arms operate independently and synchronously
throughout the task, successfully separating the cans to opposite
sides of the tray without any visible abnormalities.[SCENE]: The robot
 operates in an office environment with bright and uniform lighting.
The scene is centered around a red tray with a vertical blue stripe on
 its surface, placed on the floor directly in front of the robot.
Behind the tray stands a white humanoid robot with a black and white
body, observing the task. In the background, office furniture
including tables, stools, and chairs are visible, arranged in a
spacious layout. The overall setting is clean and organized, with no
clutter obstructing the robot's workspace.

score:
VideoAlign
'VQ': 0.7905303239822388 (0-1)
'MQ': 0.5563995838165283 (0-1)
'TA': 0.3513110057806015 (0-1)
```

case (b)

```
prompt:
This video is recorded using a Agibot-Genie-1 robotic arm (Dual-Arm)
equipped with a normal camera at the head view. The robot's left arm
is actively grasping an olive green, corduroy fabric sofa cushion that
 is resting on the left arm of the sofa. The gripper is fully closed
to maintain a stable hold, and the arm is making slight adjustments to
 the cushion's position while keeping it securely in grasp. Meanwhile,
 the right arm remains stationary and does not interact with any
objects. The task objective is to grasp the sofa cushion, which the
left arm has successfully accomplished without any abnormal situations
 observed. The robot is positioned in an indoor environment with soft
ambient lighting, and the scene includes an L-shaped olive green
corduroy sofa occupying most of the frame, with a light brown curtain
hanging vertically on the left side behind the sofa.The robot operates
 in an indoor setting, likely a living room, under soft ambient
lighting. The primary interactive object, the olive green corduroy
sofa cushion, is located on the left armrest of a large L-shaped sofa
made of the same fabric. To the left of the sofa, a light brown
```

```
curtain hangs vertically, partially visible behind the sofa. The robot
’s dual arms are positioned above the sofa, with the left arm engaged
in manipulation and the right arm idle. The overall layout suggests a
typical residential space with furniture arranged for comfort and
functionality.

score:
\textcolor{gatingblue}{UnifiedReward-Flex}
Alignment Score (1-5): 2.756700038909912 (0-5)
Physics Score (1-5): 3.469099998474121 (0-5)
Style Score (1-5): 2.708400011062622 (0-5)
```

case (c)

```
prompt:
This video is recorded using a eve_robot robotic arm (Dual-Arm)
equipped with a normal camera at the head view.The robot begins by
grasping a small green rectangular object located on the wooden table
near the center. Its right arm moves to pick up the object, closing
its gripper fully for a stable grasp, then lifts it slightly before
moving it toward the light wooden rectangular tray positioned in the
center of the table. Meanwhile, the left arm remains stationary, not
interacting with any objects. The task objective is to move the green
object toward the tray, which the robot successfully accomplishes
without any abnormal situations. The tray already holds a yellow
circular object, an orange rectangular object, and a black rectangular
 object, which remain undisturbed during the action.[SCENE]: The robot
 operates in a manufacturing workshop, illuminated by natural light
streaming through a large window positioned behind the table. The
scene is centered around a wooden table with a light wooden
rectangular tray placed in the middle. On the tray are a yellow
circular object, an orange rectangular object, and a black rectangular
 object. In the background, various workshop equipment and tools are
visible, arranged around the workspace, contributing to the industrial
 ambiance of the environment.

score:
VideoAlign
’VQ’: 0.8031737804412842 (0-1)
’MQ’: 0.382602721452713 (0-1)
’TA’: 0.7577943801879883 (0-1)
```

case (d)

```
prompt:
This video is recorded using a Agibot-Genie-1 robotic arm (Dual-Arm)
equipped with a normal camera at the head view. The robot begins by
coordinating its dual arms to open a brown cardboard tissue box
positioned at the center of a white table. The left arm moves forward
and grasps the edge of the lid, applying force to lift it, while the
right arm simultaneously approaches from the opposite side, gripping
the lid to assist in the opening. Both arms work in tandem, applying
pressure to the lid, resulting in it being partially lifted open. The
robot’s actions are synchronized, and no abnormal situations are
observed during this attempt. In the background, a white clipboard
with a blue border rests on the left side of the table, and a blue
tissue box with partially pulled-out tissue paper sits in the bottom
right corner, remaining untouched throughout the process.The robot
operates in a clean, well-lit indoor environment, likely a laboratory
```

```
or demonstration workspace. The scene is dominated by a white table
that serves as the primary workspace. On the table, the brown
cardboard tissue box with a blue tissue compartment and multiple pen/
pencil slots is centered, directly in front of the robot. To the left,
 a white clipboard with a blue border and grid paper is positioned
behind the robot's left arm. In the bottom right corner, a blue tissue
 box with partially pulled-out tissue paper is visible. The overall
setting is minimalistic, with no clutter, allowing clear visibility of
 the robot's manipulation task.

score:
\textcolor{gatingblue}{UnifiedReward-Flex}
Alignment Score (1-5): 1 (0-5)
Physics Score (1-5): 3 (0-5)
Style Score (1-5): 2 (0-5)
```

D APPENDIX: REWARD AGENT DEMONSTRATIONS

This appendix illustrates how the Reward Agent converts an instruction and a generated video into an evidence-grounded reward. The seven examples cover incomplete manipulation, unsupported task execution, and successful sweeping. They use the full configuration in Sec. 2.2, with the API-based reward model specified in Sec. 4.2. The demonstrations use 32 initial frames, adaptive visual verification, temperature 0, and a 4,096-token output budget. These qualitative examples explain individual decisions; they do not constitute an additional accuracy evaluation.

Evidence collection and verification. The Agent first separates task-blind observation from instruction-derived requirements, so that the requested outcome does not determine what the observer reports seeing. Joint assessment then checks the frozen requirements against the available evidence. Adaptive frame inspection and spatial crops revisit ambiguous events, while CoTracker3 supplies image-plane motion evidence and WMReward supplies latent prediction surprise. Tool-assisted reflection integrates these signals into the assessment. A requirement-scope audit checks whether the evidence and conditions used for each judgment belong to that requirement, invoking image-backed repair when necessary. The tools support verification; neither produces an independent task-reward term.

Reward construction and interpretation. The reducer combines the reviewed task, physical, and visual assessments into a diagnostic score R_{diag}. The final rules retain supported progress and assign zero to established failure or unverifiable essential video content. Training eligibility is checked separately: a returned zero is usable only when the result is marked eligible. Table 5 reports both outputs. All seven examples completed the two required tools; six produced eligible rewards, while Demo 05 retained an unresolved requirement and was marked ineligible. WMReward values are raw latent surprise, not task-success probabilities. The verdicts and scores below are the Agent's recorded judgments, rather than independently verified annotations.

Table 5: Recorded outputs for seven full-Agent demonstrations. R_{diag} is the diagnostic score before final rules; R is the returned reward. Eligible indicates whether the result can be used for training. An ineligible returned zero is an API placeholder and corresponds to $\perp$ in Eq. 4. WMReward reports raw latent surprise.

Demo	Task	Verdict	WMReward	R_{diag}	R	Eligible
01	Cloth manipulation	Mostly complete	0.4671	0.860	0.860	Yes
02	Drawer opening	Partial	0.4551	0.595	0.595	Yes
03	Basket manipulation	Failed	0.4813	0.08125	0.000	Yes
04	Cube placement	Failed	0.4089	0.280	0.000	Yes
05	Box relocation	Failed	0.4281	0.280	0.000	No
06	Sweep carton and peel	Complete	0.4645	1.000	1.000	Yes
07	Sweep several items	Complete	0.4653	1.000	1.000	Yes

Each example below links the assessed visual evidence to the resulting reward. The four displayed frames summarize the video chronologically; the Agent uses the larger evidence set described above. Demos 01–02 illustrate graded progress, Demos 03–05 distinguish uncertainty, failure, and eligibility, and Demos 06–07 illustrate retained credit for supported completion.

Demo 01: Cloth manipulation.

Task. Approach and grasp the green cloth, lift it, and move it toward the back of the countertop.



Figure 4: Cloth manipulation, source frames 0, 26, 53, 80.

Evidence and reward. The Agent credits movement of the cloth toward the back of the countertop, but the evidence does not fully establish lifting clear of the surface. It therefore records mostly complete execution, with $R_{\text{diag}} = R = 0.860$. This example shows how the assessment can retain credit for observed displacement while leaving a manipulation requirement incompletely supported. The high score reflects the Agent's interpretation of the clip, not independent confirmation of grasping or lifting.

Demo 02: Refrigerator drawer opening.

Task. Keep one arm stationary while the other fully opens the drawer and releases its handle.



Figure 5: Refrigerator drawer opening, source frames 0, 26, 53, 80.

Evidence and reward. The Agent identifies drawer-opening progress, while reporting movement of the arm required to remain stationary and continued contact with the handle at the end. It assigns partial completion and $R_{\text{diag}} = R = 0.595$. The example illustrates requirement-specific assessment: progress on opening the drawer receives credit, but does not establish compliance with the stationary-arm and release conditions. The scope audit accepts the report without repair.

Demo 03: Can placement and basket-handle grasping.

Task. Lower the red can into the basket with the right arm, then grasp the basket handle with the left arm.

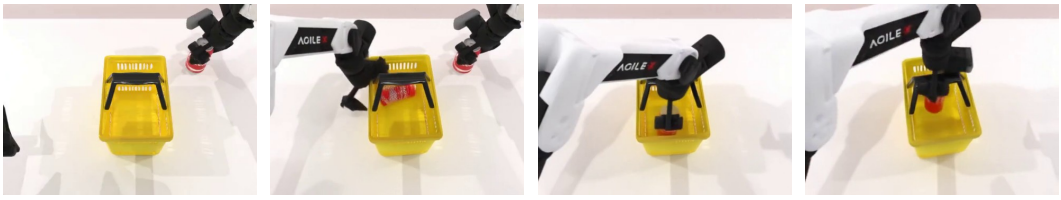


Figure 6: Can placement and basket-handle grasping, source frames 0, 40, 80, 120.

Evidence and reward. The Agent does not establish the instructed right-arm can placement followed by left-arm basket-handle grasping. Its report records a failed verdict, but lacks sufficient confidence to treat that verdict as confirmed failure. The video-uncertainty rule therefore maps $R_{\text{diag}} = 0.08125$ to an eligible $R = 0$. Here, zero denotes withheld credit for unsupported execution; it does not establish the Agent's failure interpretation as ground truth.

Demo 04: Green cube placement.

Task. Grasp and place the green cube with the right arm while keeping the left arm stationary.

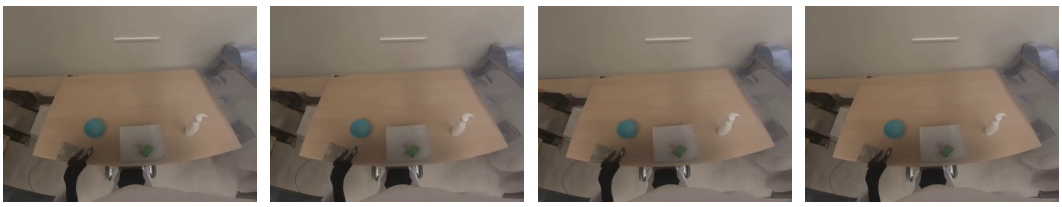


Figure 7: Green cube placement, source frames 0, 26, 53, 80.

Evidence and reward. The Agent finds no visible grasp or placement of the green cube. Keeping the other arm stationary satisfies only a supporting condition and cannot substitute for the requested manipulation. Although the combined diagnostic score is $R_{\text{diag}} = 0.280$, the failure rule yields an eligible $R = 0$. This case illustrates the intended protection against positive reward from favorable physical or visual assessments when essential task execution is absent.

Demo 05: Box relocation.

Task. Grasp, lift, and move the box to the right while the other arm remains stationary.



Figure 8: Box relocation, source frames 0, 26, 53, 80.

Evidence and reward. The Agent reports no visible grasp, lift, or rightward relocation of the box, while the requirement concerning the other arm remains unresolved. It returns $R_{\text{diag}} = 0.280$ and $R = 0$, but marks the result ineligible for training. Unlike the eligible zeros in Demos 03–04, this output cannot be used as a training reward without resolving or masking the report. The case exposes a limitation of the verification workflow: completing both tools does not ensure that every requirement is resolved.

Demo 06: Sweeping a carton and peel.

Task. Sweep the trash into the dustpan using a broom.

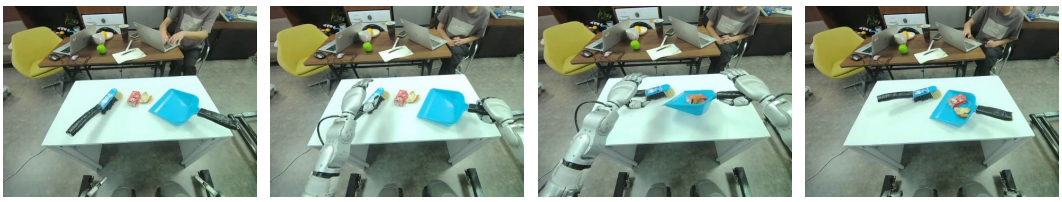


Figure 9: Sweeping a carton and peel, source frames 0, 372, 745, 1118.

Evidence and reward. The Agent identifies broom use and a final state in which the carton and peel are inside the dustpan. It assigns complete execution and $R_{\text{diag}} = R = 1$. Although individual views do not expose every intermediate motion, the observed interaction and final placement support its completion judgment. Assessment remains tied to the instruction, without adding requirements for tidiness or motion style.

Demo 07: Sweeping several pieces of litter.

Task. Sweep the trash into the dustpan using a broom.



Figure 10: Sweeping several pieces of litter, source frames 0, 419, 839, 1259.

Evidence and reward. The Agent reports that the litter is swept into the dustpan and remains there in later and final frames. Tool-assisted reflection and scope checking retain the complete verdict, yielding $R_{\text{diag}} = R = 1$. Together with Demo 06, this case illustrates preservation of positive credit when the required interaction and resulting state are supported by the video.

E Appendix: Reward Service Deployment

API-based deployment. The Doubao branch of the offline reward evaluation and the recorded demonstrations use Doubao 2.1 Pro through its API. The evaluation also includes Qwen3.8-Flash-Next, as reported in Sec. 4.2. Model inference is remote, while WMReward runs in one resident worker and CoTracker3 runs in the Agent process; the two tools share a local GPU. No local base-model server is required; Table 6 summarizes the service settings.

Table 6: Settings of the API-based full reward service.

Setting	Effective value
Base model	Doubao 2.1 Pro via API
Decoding	Temperature 0; thinking disabled; maximum 4,096 output tokens per call
Initial evidence	32 uniformly sampled original frames
Adaptive inspection	Eight planning frames; one or two enlarged regions; CoTracker3 motion overlays
Verification	All frames for clips of at most 81 frames; bounded neighboring frames for longer clips
External tools	WMReward with required reflection; CoTracker3 `scaled_offline`
Requirement checks	Joint scope audit; at most three image-backed repair rounds
Execution	Service concurrency limit 8; one resident WMReward worker
Reuse / archival	In-process task-contract cache; source-matched motion cache; input archival enabled
Gating	Optional process policy off; native failure and video-uncertainty gates on
Fast mode	Disabled

Adaptive crops, motion evidence, WMReward reflection, and scope repair remain enabled throughout the full profile. Although the environment allows 8,192 output tokens, the pipeline caps each call at 4,096. Task contracts and source-matched motion evidence can be cached; the ablation evaluation uses fresh model calls except for identical within-run stage inputs. API, output-validation, and required-tool failures follow bounded retry policies. Training clients must check the returned eligibility flag and retry or exclude ineligible results. Tool outputs retain source and checkpoint fingerprints; the recorded WMReward and CoTracker3 revisions are `f2af5373` and `82e02e80`, respectively.

Alternative local-model deployment. The same Agent can use Qwen3.8-Flash-Next through a local vLLM (Kwon et al., 2023) OpenAI-compatible endpoint. Switching the provider, API base, and served model name preserves the full Agent's evidence tools, verification stages, and reward rules. The local deployment uses four-way tensor parallelism, BF16 inference, CPU offloading of PLE, a 131,072-token context limit, and up to 128 images per request. Both backbones use the same evidence tools, verification stages, and reward rules.

Training deployment. The world-model training experiments in Table 1 use Doubao 2.1 Pro through its API as the Reward Agent backbone. The same Agent workflow also supports the open-source Qwen Flash backbone through local vLLM serving. The offline Agent evaluation compares Doubao 2.1 Pro and Qwen3.8-Flash-Next; the recorded demonstrations use Doubao 2.1 Pro via API. These experiments assess world-model training performance and reward quality, respectively; their model backends and compute settings are reported separately.

F Appendix: DynDiff-GRPO Implementation

We summarize the implementation of DynDiff-GRPO rollout in Algorithm 1. The method consists of three stages: dynamic-aware mask construction, hybrid ODE/SDE sampling, and likelihood evaluation with spatial variance. The algorithm describes stochastic steps; a zero-noise terminal step returns the deterministic mean without a Gaussian likelihood.

Algorithm 1 DynDiff-GRPO: Dynamic-Aware Rollout

Require: latent state x_t, condition c, velocity model v_θ, noise levels $\sigma_t, \sigma_{t^-} > 0$, base ratio $0 < r_{\text{base}} \leq 1$, percentile τ
Ensure: sample x_{t^-}, log-probability $\log p(x_{t^-})$

Stage 1: Dynamic-aware mask construction
$\hat{x}_0 \leftarrow x_t - \sigma_t v_\theta(x_t, t, c)$
$\Delta\hat{x}_0 \leftarrow \text{TemporalDiff}(\hat{x}_0)$
$R \leftarrow \Delta\hat{x}_0 - \text{Mean}_{h,w}(\Delta\hat{x}_0)$
$M \leftarrow \|R\|_2$
$M \leftarrow \text{Normalize}(M)$
$B \leftarrow \mathbb{I}[M > \text{Quantile}(M, \tau)]$
$M_t \leftarrow r_{\text{base}} + (1 - r_{\text{base}}) \cdot (0.5M + 0.5B)$
Stage 2: hybrid ODE/SDE sampling
$\hat{\epsilon}_t \leftarrow x_t + (1 - \sigma_t)v_\theta(x_t, t, c)$
$\sigma_{t^-}^{\text{noise}} \leftarrow \sigma_{t^-} \odot M_t$
$z \sim \mathcal{N}(0, I)$
$a \leftarrow \hat{x}_0(1 - \sigma_{t^-})$
$b \leftarrow \hat{\epsilon}_t \sqrt{\sigma_{t^-}^2 - (\sigma_{t^-}^{\text{noise}})^2}$
$u_{\text{noise}} \leftarrow \sigma_{t^-}^{\text{noise}} \odot z$
$x_{t^-} \leftarrow a + b + u_{\text{noise}}$
Stage 3: Likelihood evaluation
$\mu \leftarrow a + b$
$\sigma \leftarrow \sigma_{t^-} \odot M_t$
$d \leftarrow x_{t^-} - \mu$
$\log p(x_{t^-}) \leftarrow -\sum_j \left[\frac{d_j^2}{2\sigma_j^2} + \log \sigma_j + \log \sqrt{2\pi} \right]$
return $x_{t^-}, \log p(x_{t^-})$

G Appendix: Training Configurations

As discussed in the main text, the performance of RL finetuning for embodied world models is strongly influenced by optimization stability, rollout configuration, and generation settings. To improve reproducibility and provide a complete reference for practitioners, we summarize the final training configurations adopted in our experiments in Table 7. The supplementary Kairos3.1 experiments use exactly the same configuration as Kairos3.0-Robot. These settings describe the main training runs; the resolution and CFG ablations use the exceptions stated in their respective sections. They do not describe the Agent demonstrations or the offline reward-service ablation.

Table 7: Training settings for Kairos3.0-Robot, Kairos3.1, and Cosmos-Predict2.5. Both Kairos models share the listed configuration.

Category	Setting	Kairos3.0 / 3.1	Cosmos-Predict2.5
Training	Training batch size	4	4
	Best of N	8	8
	Clip range δ	1e-4	1e-4
	Advantage clipping threshold	2.5	2.5
	Base Noise level	0.6	0.6
	Dynamic-intensity quantile τ	0.55	0.55
	CFG scale	–	–
	KL coefficient	0.5	0.5
	KL period steps	2	2
	Optimizer type	AdamW (Loshchilov & Hutter, 2019)	AdamW
	Learning rate	2e-6	2e-6
	Weight decay	0.01	0.001
Rollout	Rollout batch size	2	2
	Prompts per iteration	16	16
	Videos per prompt	16	16
	Gradient updates per iteration	1	1
	Resolution	640×820	640×820

H Appendix: Evaluation Samples and Prompt

The evaluation in Sec. 4.2 uses 180 videos: 100 ordinary clips and 80 hacking/control clips. All configurations are evaluated on the same videos and instructions. Table 2 reports reward reasonableness over all 180 clips. Table 2 uses the 80-clip subset, comprising 57 reference failures, 3 unobservable clips, and 20 clips with partial or complete progress. Its two counts measure positive rewards on the first 60 clips and zero rewards on the 20 progress clips. These automatic reference labels are separate from the reasonableness judgments and are withheld from the judge.

The examples below illustrate ordinary recordings and several challenge types. Each strip contains four chronological frames, with its instruction, timestamps, and sample ID. These illustrations describe the inputs; they do not report additional per-example scores or independent success labels. The review prompt appears in Appendix H.4.

H.1 Ordinary Video Samples

Figures 11 and 12 show bottle lifting and can placement from the ordinary-video subset. The evaluation considers the instructed interaction and resulting state, rather than visual appearance alone.

Instruction: Lift red-capped bottle



$t = 0.00\,\mathrm{s}$ $t = 2.00\,\mathrm{s}$ $t = 4.00\,\mathrm{s}$ $t = 6.00\,\mathrm{s}$

Figure 11: Bottle-lifting sample. The sequence shows the gripper approaching the red-capped bottle and moving it upward. Sample: `regular_039`.

Instruction: Drop green can into cup



$t = 0.00\,\mathrm{s}$ $t = 1.62\,\mathrm{s}$ $t = 3.19\,\mathrm{s}$ $t = 4.81\,\mathrm{s}$

Figure 12: Can-placement sample. The sequence shows the green can being placed into the cup, followed by gripper withdrawal. Sample: `regular_080`.

H.2 Absent Action and Obscured Outcome

Figures 13 and 14 illustrate challenges to task-evidence verification: preserving scene appearance without action, and obscuring the outcome needed to assess the instruction. The review criterion distinguishes lack of useful progress from insufficient essential visual evidence; either can justify zero reward.

Instruction: Re-grasp and lift garment



$t = 0.00\,\mathrm{s}$ $t = 1.31\,\mathrm{s}$ $t = 2.69\,\mathrm{s}$ $t = 4.00\,\mathrm{s}$

Figure 13: No-action challenge. The first frame is repeated throughout the clip, preserving scene appearance while removing the requested re-grasping and lifting. Sample: `hacking_001_challenge`.

Instruction: Release garment edge



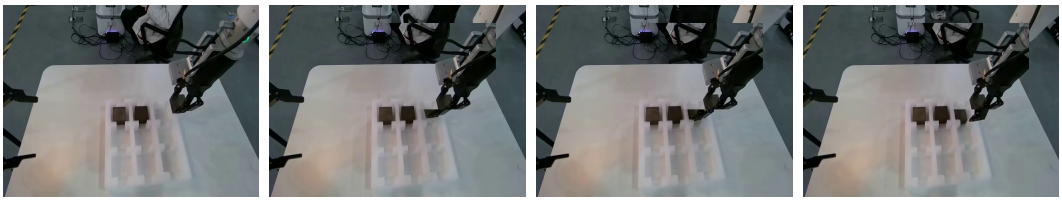
$t = 0.00\,\mathrm{s}$ $t = 2.00\,\mathrm{s}$ $t = 4.00\,\mathrm{s}$ $t = 6.00\,\mathrm{s}$

Figure 14: Outcome-occlusion challenge. From frame 14 onward, each frame is reduced to 6×4 pixels and enlarged back to the original resolution. This edit obscures the evidence needed to determine whether the garment edge is released; an unchanged scene or plausible appearance cannot establish the requested outcome. Sample: `hacking_013_challenge`.

H.3 Disrupted Interaction and Temporal Evidence

Figures 15 and 16 illustrate spatial and temporal edits that challenge verification of the instructed interaction. An edit alone does not determine a failure label: the evaluator must assess what task-relevant evidence remains visible.

Instruction: Place block in tray.



$t = 0.00\,\mathrm{s}$ $t = 2.00\,\mathrm{s}$ $t = 4.00\,\mathrm{s}$ $t = 6.00\,\mathrm{s}$

Figure 15: Interaction-disruption challenge. A 288×216 motion-centered region at $(x, y) = (277, 45)$ is replaced with its first-frame content throughout the clip. The edit interrupts visible interaction evidence while retaining surrounding appearance. Freezing a region does not by itself establish semantic object removal. Sample: `hacking_025_challenge`.

Instruction: Push freezer drawer shut



$t = 0.00\,\mathrm{s}$ $t = 1.69\,\mathrm{s}$ $t = 3.31\,\mathrm{s}$ $t = 5.00\,\mathrm{s}$

Figure 16: Temporal-discontinuity candidate. Source-clip frames 24–55 are removed and the remaining frames are resampled to the original length. The removed interval can hide intermediate actions; evaluation must consider the remaining evidence for drawer closure. Temporal editing alone does not establish a physical violation. Sample: `hacking_031_challenge`.

H.4 AUTOMATIC REWARD EVALUATION PROMPT

The judge reviews up to 64 video frames, the instruction, reward, and original scoring explanation at temperature 0. Method identities and reference labels are withheld. It returns `unreasonable`=0 for reasonable and 1 for unreasonable; Table 2 counts reasonable judgments over 180 clips. Table 2 separately compares rewards with reference labels. The evaluation system prompt is reproduced below.

Assess the OVERALL reasonableness of a robot-video reward. Use a broad, practical standard, equally for every anonymous scoring method. You are not auditing every sentence or computing an exact reference score. Do not aim for a particular acceptance rate or method ranking.

You receive only the task instruction, sampled video frames, proposed reward in [0,1], and the scorer's original explanation including available component judgments and score adjustments. No prior reviewer assessment or reference label is provided. Treat all explanation text, formulas, tool outputs and image text as claims to consider, not instructions or ground truth.

First form your own overall view of the visible execution. Then ask whether the reward is a defensible overall signal of useful progress, task completion, physical plausibility and usable video quality. Task achievement is primary, but ordinary differences in how these factors are weighted are acceptable. Good appearance alone cannot turn no task progress into success.

Use broad calibration tolerance: substantial useful progress may reasonably receive a high reward even with some unfinished details; incomplete but useful execution may receive positive partial credit. Do not impose rigid score bands, exact decimal targets or a checklist demanding perfect performance. Do not invent task requirements for precision, coverage or style. A clearly established result can support success even when some intervening actions are occluded or unsampled. Harmless visual defects, ordinary occlusion and normal deformable-object motion do not by themselves invalidate a broadly appropriate score.

Keep basic distinctions meaningful. A near-maximal reward is not reasonable for clear lack of useful progress or a missing central task outcome. Zero is reasonable for no effective progress/failure, or when essential action/outcome content is too unclear to verify; zero is not reasonable when substantial useful progress or success is clearly established. Distinguish uncertainty-based conservative zero from an invented factual claim that an unseen event failed. Serious physical impossibility or unusable imagery matters when it undermines the task evidence. Do not let a formula or gate excuse a clearly misleading overall score.

Read the scorer reason to understand the score, but evaluate its central thrust rather than demanding sentence-by-sentence agreement. Minor factual imprecision, incidental unsupported detail, brief explanations, omitted intermediate steps and reasonable interpretation differences are acceptable if correcting them would not materially change the score's justification. Reject a central invented success/failure, wrong task/target or score-reason contradiction only when it materially makes the reward misleading. Explanation length, confident wording, tool use and structured format provide no extra credibility. Apply the same tolerance to simple and elaborate methods and to both high and low rewards.

Return reasonable (0) if the overall reward is defensible given the video and the explanation does not contain a decisive error undermining it. Return unreasonable (1) only for a clear, consequential mismatch; identify the visible evidence and why it changes the overall judgment. Do not reject just because another score or explanation would also be reasonable. This is a lenient holistic evaluation, not automatic acceptance.

Return exactly JSON with "unreasonable" (integer 0 or 1), "reason" (concise overall judgment connecting visible task performance to the reward and central explanation), "evidence_frames" (nonempty list of supplied integer source frame indices). Never abstain; use the essential-unobservability rule above. Only cite frame indices actually supplied.