

Persistent Robot World Models: Stabilizing Multi-Step Rollouts via Reinforcement Learning

Jai Bardhan , Patrik Drozdik , Josef Sivic , and Vladimir Petrik 

Czech Institute of Informatics, Robotics and Cybernetics, Czech Technical University
in Prague

`{first.last}@cvut.cz`

<https://www.jaibardhan.com/persistworld>

Abstract. Action-conditioned robot world models generate future video frames of the manipulated scene given a robot action sequence, offering a promising alternative for simulating tasks that are difficult to model with traditional physics engines. However, these models are optimized for short-term prediction and break down when deployed autoregressively: each predicted clip feeds back as context for the next, causing errors to compound and visual quality to rapidly degrade. We address this through the following contributions. First, we introduce a reinforcement learning (RL) post-training scheme that trains the world model on its own autoregressive rollouts rather than on ground-truth histories. We achieve this by adapting a recent contrastive RL objective for diffusion models to our setting and show that its convergence guarantees carry over exactly. Second, we design a training protocol that generates and compares multiple candidate variable-length futures from the same rollout state, reinforcing higher-fidelity predictions over lower-fidelity ones. Third, we develop efficient, multi-view visual fidelity rewards that combine complementary perceptual metrics across camera views and are aggregated at the clip level for dense, low-variance training signal. Fourth, we show that our approach establishes a new state-of-the-art for rollout fidelity on the DROID dataset, outperforming the strongest baseline on all metrics (e.g., LPIPS reduced by 14% on external cameras, SSIM improved by 9.1% on the wrist camera), winning 98% of paired comparisons, and achieving an 80% preference rate in a blind human study.

Keywords: World Models · Reinforcement Learning · Robotics

1 Introduction

Action-conditioned video diffusion world models (WM) represent a transformative frontier for robot learning, offering the potential to simulate complex, human-centric tasks that are notoriously difficult to model with traditional physics-based engines. Recent work [9] has demonstrated that finetuning pre-trained video diffusion backbones with action conditioning on large-scale robotics datasets [22] can yield world models that generate impressively faithful clips that



Fig. 1: Autoregressive video rollout quality. We use an action-conditioned robot world model to generate multi-view image predictions from a single observed state. The ground-truth (blue) is compared against the baseline (red) and our post-trained model **PersistWorld** (green). While the baseline accumulates error and destroys the object (cyan bowl) within seconds, our method maintains structural integrity and spatial consistency, establishing a new state-of-the-art in rollout fidelity.

are consistent with the robot’s actions. By synthesizing high-fidelity visual rollouts conditioned on robot actions, these models can serve as scalable virtual environments for benchmarking and improving Vision-Language-Action (VLA) policies.

However, realizing this potential requires generating *long-horizon* autoregressive rollouts — multiple seconds of coherent video — where each predicted clip feeds back as context for the next. This is precisely where current models break down. This phenomenon, known as *exposure bias* [31], arises from a train/test distribution mismatch, i.e., the model is trained to predict from *ground-truth* history frames, but at deployment it must condition on its own previously generated outputs, which carry growing imperfections. The consequences are rapid and severe. Within seconds of autoregressive generation, manipulated objects lose their structural identity — a bowl dissolves into an amorphous blob (see Fig. 1) — robot end-effectors drift from their commanded trajectories, and entire scene configurations decohere (see Fig. 3).

Reinforcement learning (RL) offers a natural framework for addressing this problem: by computing a training signal directly on the model’s own autoregressive rollouts, it incentivizes consistent long-horizon generation rather than single-step accuracy. However, applying RL to diffusion models is challenging — standard policy-gradient methods require likelihoods that diffusion models do not provide, and backpropagating through the full denoising process is prohibitively expensive. Recent work [42] offers an elegant workaround by generating multiple candidate outputs, scoring them with a reward, and using the comparison to update the model — avoiding backpropagation through denoising entirely. However, this approach was developed for single-image generation and does not directly apply to our setting for two reasons. First, our robot world model uses a different type of denoising network, which means the theoretical guarantees need to be re-derived. Second, in image generation, the model can simply draw multiple independent samples from the same text prompt and compare them. In autoregressive video, there is no fixed prompt — each generation step builds on the previous output, creating an evolving shared state. This means we need a new mechanism for producing comparable candidates that can be meaning-

fully ranked against each other. We address these challenges via the following contributions:

1. **RL post-training for robot world models.** We introduce a post-training scheme that optimizes the world model directly on its own autoregressive rollouts rather than on ground-truth histories. We adapt a recent contrastive RL method for diffusion models [42] to our setting, where the denoising network directly predicts clean frames rather than intermediate noise, and show that convergence guarantees carry over exactly (Sec. 3.2).
2. **A training protocol for autoregressive robot world models.** Autoregressive robot world model has no fixed prompt from which multiple candidates can be drawn and compared. We observe that the model’s accumulated history at any rollout step serves as a natural shared context from which multiple candidate continuations can be independently generated and ranked. By randomly sampling how deep into the rollout we branch these candidates, we expose training to both mild early-stage and severe late-stage error regimes (Sec. 3.3).
3. **Multi-view visual rewards and task-relevant evaluation.** We design clip-level rewards that combine complementary perceptual metrics across all three camera views, normalized so that the training signal reflects relative quality within each group of candidates. We further introduce object-centric and robot-centric masked evaluations that confirm our improvements come from better modeling of task-relevant dynamics rather than background preservation (Sec. 3.4 and Sec. 4).
4. **State-of-the-art rollout quality.** Our post-trained model sets a new state-of-the-art on the DROID dataset [22] across all visual quality metrics, with the largest improvements on the wrist camera—the view most critical for capturing fine-grained object manipulation. In paired comparisons, our model outperforms the baseline on approximately 98% of validation samples. A blind human preference study confirms these gains, with raters favoring our rollouts 80% of the time (Sec. 4).

2 Related work

Video Diffusion Models. Diffusion models have emerged as a dominant paradigm for high-fidelity visual generation and have been successfully extended from images to videos by modeling space-time volumes with denoising objectives [14, 15, 34]. Latent video diffusion approaches adapt image diffusion backbones with temporal modules, enabling strong image-to-video and text-to-video generation while encoding rich visual priors about object motion, lighting, and physical plausibility [4]. While these models produce impressive open-loop clips, long-horizon generation via autoregressive stitching of short segments causes errors in early frames to compound as the model conditions on its own imperfect outputs, leading to temporal drift and degradation. Our setting inherits this challenge in an action-conditioned, multi-view robotics regime.

Robotic World Models and Video-Based Planning. World models have a long history in model-based RL as learned dynamics models enabling planning through imagined rollouts [10–13]. Several works frame robot planning and evaluation directly as video generation [3, 6, 17, 23], and IRASim [44] demonstrates that a trajectory-conditioned diffusion world model can serve as a policy evaluation. Large pre-trained video diffusion backbones have been adapted into controllable robot world models via action conditioning: Ctrl-World [9] and WPE [30], both trained on DROID [22], generate multi-view manipulation trajectories and can rank downstream policy performance; AVID [33] adapts pretrained video diffusion via a learned mask adapter without parameter access; and UWM [43] jointly models video and action diffusion. None of these directly address the training-inference mismatch under self-conditioned rollout.

Post-Training Diffusion Models. Aligning diffusion models to downstream objectives has been studied through RL and preference optimization [2, 24, 29, 35, 39]. DiffusionNFT [42] proposes a negative-aware fine-tuning objective on the forward diffusion process, enabling efficient online RL updates without back-propagating through the denoising trajectory. DPPO [32] applies RL fine-tuning to diffusion-based robot action policies; our work instead post-trains the world model itself. RLVR-World [38] applies RL with verifiable rewards to improve world model transition quality, further evidencing that RL objectives outperform MLE for rollout fidelity. However, they work with token based models, whereas we work on improving video diffusion models. Contemporary to our work, [36] applies contrastive RL post-training [42] to a camera-pose-conditioned world model, targeting improvements in camera pose following and visual fidelity—including a prefix rollout strategy in which the model’s own outputs serve as context for subsequent clip generation. Our work shares this core motivation and training paradigm, but differs in several respects. We adopt a randomized prefix horizon during post-training rather than a fixed schedule, which we find better captures the distribution of compounding errors at test time. Our setting targets dynamic manipulation scenes in a multi-view robotic world model and we introduce a novel adaptation of the contrastive RL objective to the $\mathbf{x}_0$ -prediction parameterization used by some robot WMs. We further design visual rewards that are efficient and scalable for the multi-view robot manipulation setup. We find that compared to the HPSv3 [26] reward that is used by [36], our rewards result in much better performance. Finally, we additionally validate our approach through quantitative evaluation on robot-centric metrics and a human preference study.

Exposure Bias and the Rollout Gap. The mismatch between teacher-forced training and self-conditioned inference is a longstanding problem in sequence generation [1, 31], and has been studied specifically for diffusion models [27]. In video generation the effect is amplified: small per-frame errors accumulate over long rollouts, degrading coherence and limiting world model utility for evaluation and simulation. We address this *rollout gap* with RL post-training that directly exposes the model to its own generated histories during training.

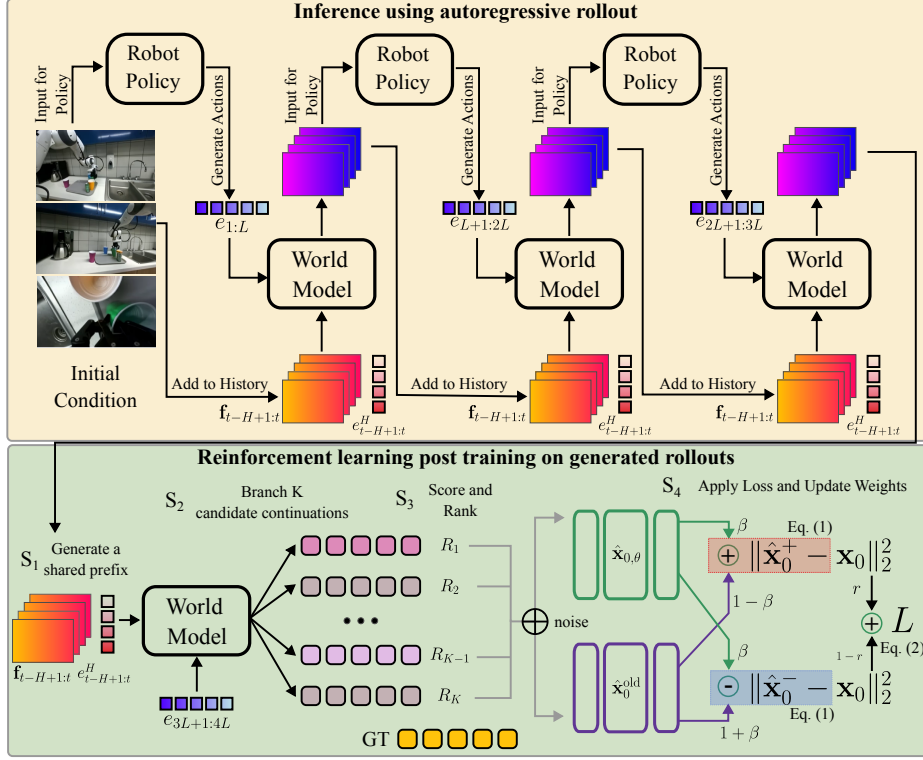


Fig. 2: Overview of our method: (*Top*) **Autoregressive inference:** A robot policy generates actions fed to the world model, which produces multi-view frames that are appended to the history buffer and condition the next generation step. (*Bottom*) **RL post-training:** (**S1**) A shared variable-length prefix is rolled out autoregressively from a ground-truth initial condition. (**S2**) K independent candidate continuations are branched from the frozen prefix state. (**S3**) Candidates are scored against ground-truth using multi-view perceptual rewards. (**S4**) Reward weights r scale implicit positive/negative $\mathbf{x}_0$ predictions used in contrastive model updates via loss L .

Large-Scale Robot Datasets. Large-scale robot datasets enable both training data-hungry video world models and the held-out ground-truth trajectories our reward computation relies on. DROID [22] provides diverse multi-camera manipulation demonstrations, and Open X-Embodiment [28] aggregates demonstrations across many embodiments and institutions—making dataset-driven reward evaluation feasible without human preference labels.

3 Improving Robot World Models with Reinforcement Learning

Action-conditioned video diffusion world models are trained to predict the next chunk of video frames given a clean, ground-truth history — a setup that works

well in isolation, but breaks down the moment the model is deployed autoregressively over longer horizons. The central challenge we address is the following: because the model has never seen imperfect history at training time, any error introduced at one step propagates forward and compounds at the next, causing rollout quality to degrade rapidly. Resolving this requires more than better data or longer training; it requires a different approach to training — one that explicitly optimizes the model under the same auto-regressive, self-conditioned regime in which it operates at test time.

This section develops our approach in three stages. We begin by characterizing the train/test distribution mismatch and why standard training cannot resolve it (Sec. 3.1). We then formulate post-training as an online reinforcement learning (RL) problem and derive a tractable objective by adapting a contrastive forward-process training designed for velocity-prediction flow-matching models [42] — to the EDM [21] x_0 -prediction parameterization of Ctrl-World’s SVD backbone; we show that the branch construction and policy-improvement guarantees carry over exactly, with no σ -dependent correction terms (Sec. 3.2). Finally, we address two concrete design challenges: how to construct a group-relative training signal from autoregressive video rollouts (Sec. 3.3), and how to define rewards that faithfully assess multi-view, multi-step visual quality (Sec. 3.4).

3.1 The Closed-Loop Gap in Autoregressive World Models

Model overview. An autoregressive robot world model, such as Ctrl-World [9], operates in a loop: at each step, it receives a history buffer of recent frame latents encoding the visual context so far, together with past and future robot actions, and generates the next chunk of future frames. The generated frames are then encoded and appended to the history buffer, which conditions the next generation step. In detail, at each autoregressive step, the model receives three inputs: (i) a history buffer $\mathbf{f}_{t-H+1:t}$ of $H=6$ recent frame latents that encode the visual context of what has transpired so far; (ii) the corresponding robot end-effector (EEF) poses $\mathbf{e}_{t-H+1:t}^H$; and (iii) a sequence of future EEF pose targets $\mathbf{e}_{t+1:t+L} \in \mathbb{R}^{L \times 7}$. From these, the model simultaneously generates $L=5$ future frames across three camera views — two external views and one wrist-mounted view.

The closed-loop gap. Training follows a standard diffusion objective under teacher forcing: given a clip of length $H + L$ from the dataset, the model learns to denoise the final L frames conditioned on the preceding H *ground-truth* frames as history. This produces a reliable training signal, but installs a structural mismatch with deployment. At test time, no ground-truth history is available; each generated clip is encoded and appended to the rolling history buffer, which then conditions the next denoising pass. The model must now condition on its own previous outputs — inputs it was never trained to handle. The consequence is an error compounding loop. A minor spatial or temporal inaccuracy in clip t

corrupts the latents stored in the history buffer. These corrupted latents condition clip $t + 1$ with increased error, which in turn corrupts the history for clip $t + 2$. Within seconds, generated scenes decohere: object configurations blur, robot state diverges from the commanded trajectory, and scene identity dissolves. Rollouts beyond a few seconds become unreliable as surrogates for real-world execution — precisely the use-case that makes world models valuable. This is not a data sufficiency problem. No amount of teacher-forced training gives the model incentive to be robust to its own imperfect history, because imperfect history is absent from the training distribution by construction. What is needed is a training signal computed directly from the model’s own autoregressive outputs — one that rewards coherent closed-loop generation and penalizes compounding drift.

3.2 Online RL Post-Training via Reward-Contrasted Denoising

Online reinforcement learning offers a principled solution: generate rollouts autoregressively, evaluate them against held-out ground truth, and update the model toward higher-fidelity outputs — with the training distribution defined by the model’s own production rather than teacher-forced ground truth. Because the reward is computed on self-generated frames, the training signal inherently reflects the closed-loop statistics of deployment. The challenge is making this compatible with diffusion models.

Reward-conditioned forward-process training. We address this challenge by recasting policy improvement as contrastive denoising [42]: rather than estimating reverse-process likelihoods, we generate a group of candidate outputs, score them with a reward, and encode the relative quality signal directly into the denoising loss — reinforcing what the model produces for high-reward candidates and penalizing what it produces for low-reward ones. The contrastive denoising approach [42] was originally derived for velocity-prediction flow-matching models such as SD3 [8]. However, some world models — including Ctrl-World [9], which we build upon — employ an x_0 -prediction parameterization, where the network directly estimates the clean data x_0 rather than a velocity field. We show that the contrastive denoising framework transfers naturally to this setting: because the mapping from network output to clean-data prediction is affine, the contrastive objective construction and its policy-improvement guarantees carry over exactly, with no additional correction terms.

The full derivation is in the **Appendix A**. The resulting x_0 -adapted objective takes the following form. Let $\hat{\mathbf{x}}_{0,\theta}$ denote the model’s clean data (x_0) prediction and $\hat{\mathbf{x}}_0^{\text{old}}$ a frozen exponential moving average (EMA) copy serving as the reference policy. For a candidate with normalized reward weight $r \in [0, 1]$ and mixing coefficient β , we construct implicit positive and negative clean data (x_0) predictions:

$$\hat{\mathbf{x}}_0^+ = (1 - \beta) \hat{\mathbf{x}}_0^{\text{old}} + \beta \hat{\mathbf{x}}_{0,\theta}, \quad \hat{\mathbf{x}}_0^- = (1 + \beta) \hat{\mathbf{x}}_0^{\text{old}} - \beta \hat{\mathbf{x}}_{0,\theta}, \quad (1)$$

and minimize the reward-weighted denoising loss:

$$\mathcal{L}(\theta) = \mathbb{E} \left[r \|\hat{\mathbf{x}}_0^+ - \mathbf{x}_0\|_2^2 + (1 - r) \|\hat{\mathbf{x}}_0^- - \mathbf{x}_0\|_2^2 \right]. \quad (2)$$

Intuitively, the difference $\hat{\mathbf{x}}_{0,\theta} - \hat{\mathbf{x}}_0^{\text{old}}$ defines the direction in which the current model has drifted from the frozen reference. The positive branch $\hat{\mathbf{x}}_0^+$ *extrapolates* along this direction: it takes the reference prediction and moves it toward the current model by a factor β , amplifying whatever changes the model has learned. Conversely, the negative branch $\hat{\mathbf{x}}_0^-$ *reverses* this direction, constructing a counterfactual prediction that moves away from the current model’s output. The loss in Eq. 2 then uses the reward weight r to interpolate between fitting $\hat{\mathbf{x}}_0^+$ (reinforcing the model’s current direction for high-reward samples) and fitting $\hat{\mathbf{x}}_0^-$ (repelling the model from its own predictions for low-reward samples). The mixing coefficient β controls the strength of this amplification: larger β produces a stronger reinforcement signal but risks destabilizing training, while $\beta \rightarrow 0$ recovers standard supervised denoising against the reference. Note that this formulation requires only the clean generated samples and the reference predictions — it avoids backpropagating through the denoising chain entirely, making it compatible with any black-box sampler.

3.3 Adapting Group-Relative Training to Autoregressive Video

The RL formulation above addresses how to update a diffusion model given reward-scored samples. Applying it to an autoregressive world model raises a second, distinct challenge: the formulation assumes a natural grouping structure — a shared conditioning input from which multiple independent candidate outputs are drawn. In image generation, this structure is straightforward: sample K independent images from the same text prompt and compare them by reward. In autoregressive video generation, no such fixed prompt exists. Each generation step produces a clip that modifies the shared history buffer, which then conditions the next step; candidate clips are not independent draws from a common condition, but sequential extensions of an evolving shared state.

The key observation is that the history buffer state immediately before any generation step plays exactly the role of the prompt in the group-relative setting: it is the accumulated context from which distinct candidate continuations can be independently branched. Freezing this buffer state and sampling K independent candidate next clips from it yields a group that shares a common context, enabling meaningful reward-based comparison and contrastive training. This recovers the shared-context / independent-response structure required by group-relative objectives. We realize this structure through the following rollout protocol at each training step:

S₁: Generate a shared prefix. Starting from a single ground-truth observation — with the history buffer backfilled by replicating its encoded latent — we autoregressively generate P consecutive clips, feeding the model’s own outputs back as history at each step. This mirrors closed-loop deployment and

produces a history buffer state that has been corrupted by the model’s own accumulated errors. The prefix length is sampled as $P \sim \text{Unif}\{0, 1, \dots, 9\}$, exposing training to the full spectrum of rollout positions — from early steps where the buffer is nearly clean to late steps where compounding drift is severe.

- S₂: **Branch K candidate continuations.** From the frozen prefix history buffer, we independently sample $K=16$ candidate next segments. Each candidate is a short autoregressive sequence of F chunks: the model generates L frames across all three views simultaneously, encodes and appends them to a private copy of the history buffer, and repeats for F steps. Each candidate follows its own distinct stochastic trajectory from the shared context.
- S₃: **Score and rank.** A visual reward $R_t^{(k)}$ is computed for each candidate by comparing its generated frames against held-out ground-truth frames across all three camera views (Sec. 3.4). Rewards are group-normalized over the K candidates to form relative advantages, removing the influence of absolute reward scale at different rollout positions.
- S₄: **Update the model.** The group-normalized reward weights are used to scale the positive and negative denoising losses (Eq. 2), and the model is updated via gradient descent. Only LoRA adapters and the action encoder receive gradient updates; the backbone is frozen.

The variable prefix length serves two purposes. It ensures the model is optimized to maintain quality across the full rollout depth, not only at short horizons. It also exposes the update to diverse history buffer corruption profiles — from lightly drifted early-step buffers to heavily degraded late-step ones — preventing overfitting to any single error regime.

3.4 Visual Rewards for Multi-View Video Clips

Defining an effective reward for autoregressive robot video generation requires carefully considering what to measure and how to aggregate it. The signal must be dense enough to be informative at each training step, directly tied to perceptual quality rather than proxy statistics, and consistent across the three camera views, which provide complementary perspectives on the manipulated scene. A reward aggregated across an entire long-horizon rollout would be high-variance and would make credit assignment to specific generations difficult; we therefore score at the granularity of individual clips.

For each candidate clip at time t , we compare generated frames $\hat{\mathbf{x}}_{t+1:t+L}^{(v)}$ against the held-out ground-truth frames $\mathbf{x}_{t+1:t+L}^{(v)}$ for each view $v \in \mathcal{V} = \{\text{wrist}, \text{ext}_1, \text{ext}_2\}$. Per-frame metrics are first averaged temporally over the L frames of the clip:

$$\bar{m}_t^{(v)} = \frac{1}{L} \sum_{\ell=1}^L m\left(\hat{\mathbf{x}}_{t+\ell}^{(v)}, \mathbf{x}_{t+\ell}^{(v)}\right), \quad m \in \{\text{LPIPS}, \text{SSIM}, \text{PSNR}\}, \quad (3)$$

and then averaged equally across the three views:

$$\bar{m}_t = \frac{1}{3} \left(\bar{m}_t^{(\text{wrist})} + \bar{m}_t^{(\text{ext}_1)} + \bar{m}_t^{(\text{ext}_2)} \right). \quad (4)$$

We use three complementary metrics to capture distinct failure modes. LPIPS [41] measures perceptual similarity in deep feature space, penalizing structural distortions even when pixel values are numerically close. SSIM [37] captures luminance, contrast, and local structural fidelity over spatial patches. PSNR [19] provides a global signal-to-noise measure that is sensitive to large pixel deviations, acting as a coarse indicator of catastrophic scene drift. Using all three produces a reward robust to the blind spots of any individual metric. LPIPS captures perceptual distortions invisible to pixel-level metrics, SSIM is sensitive to local structural changes, and PSNR flags large-scale pixel drift; combining them guards against failure modes that any single metric would miss. The per-view, per-metric averages are combined into a single scalar reward:

$$R_t = -w_{\text{LPIPS}} \overline{\text{LPIPS}}_t + w_{\text{SSIM}} \overline{\text{SSIM}}_t + w_{\text{PSNR}} \overline{\text{PSNR}}_t, \quad (5)$$

where LPIPS is negated (it is lower-better), and the weights w are set to bring the three components to a comparable numerical scale (values in Sec. 4).

Group normalization. Because absolute reward values vary significantly across rollout positions — early clips score much higher than late ones — we normalize rewards within each group to focus the training signal on relative quality differences. The per-candidate rewards $R_t^{(k)}$ are group-normalized over the K candidates via z-score normalization:

$$A^{(k)} = \frac{R^{(k)} - \mu_R}{\sigma_R + \epsilon}, \quad \mu_R = \frac{1}{K} \sum_{k=1}^K R^{(k)}, \quad \sigma_R = \text{std}_k(R^{(k)}). \quad (6)$$

This converts absolute reward values into relative rankings within the group, removing the confound of reward scale variation across rollout positions. The z-scored advantages are clipped to $[-1, 1]$ and linearly rescaled to the $[0, 1]$ range required by Eq. 2:

$$r^{(k)} = \frac{\text{clip}(A^{(k)}, -1, 1) + 1}{2}. \quad (7)$$

This normalization encourages the model to discriminate between better and worse continuations from the same context, and keeps the gradient magnitude bounded regardless of the absolute level of visual quality at any given rollout position.

4 Experiments

Implementation details. We use the pre-trained Ctrl-World model [9] as our base. For the proposed post-training, we apply LoRA [16] adapters to the UNet

backbone (rank $r=64$, $\alpha=64$) and additionally finetune the action encoder; all other parameters (other UNet layers, VAE, etc.) are frozen. We train for 6,000 steps with learning rate 1×10^{-4} using the Muon optimizer [20] with the batch size 64 and group size $K=16$. Additionally, we subsample the group elements by taking the 10 most informative samples (top-5 and bottom-5 ordered by the reward) per update step. Reward weights are set to: $w_{\text{LPIPS}}=w_{\text{SSIM}}=1$ and $w_{\text{PSNR}}=\frac{1}{32}$, with the $\frac{1}{32}$ factor bringing PSNR into a comparable numerical range with $\text{SSIM} \in [0, 1]$. From our ablations this leads to the best average performance across all metrics. The model is trained on 8 NVIDIA H200 GPUs for 2 days. Primary training cost is the autoregressive rollout generation, common in prior works [18, 25]. Overall, we found that our choices are a conservative default and one can tune even more to obtain better performance; for example, see Table 6 in Appendix. For more details on ablations, see Appendix B.

Dataset. We evaluate on the DROID dataset [22], a large-scale robot manipulation dataset collected on a Franka Emika Panda robot across a diverse set of tabletop environments. DROID comprises teleoperated demonstrations across a wide variety of everyday manipulation tasks and uses a standardized three-camera setup (two external cameras and one wrist-mounted camera). We use Ctrl-World’s held-out validation split for all quantitative evaluations.

Autoregressive rollout quality evaluation. We evaluate autoregressive rollout quality on pre-recorded trajectories from the validation split. Starting from a single observed state (frames from all cameras + robot EEF pose), we generate 14 consecutive clips ($14 \times L=70$ frames, covering approximately 11s at 5 Hz) using the autoregressive procedure from Sec. 3.3. We then compare the generated frames to the corresponding ground-truth frames from the dataset using SSIM [37], PSNR [19], and LPIPS [41]. We report metrics separately for external cameras and the wrist camera, as these views capture qualitatively different aspects of the scene.

Our model establishes a new state-of-the-art for autoregressive rollout quality on the DROID dataset, consistently outperforming all baselines—WPE [30], IRASim [44], and Ctrl-World [9]—across every metric (Table 1). Compared to Ctrl-World baseline, our approach achieves significant gains on external cameras, improving PSNR by 1.40 dB and reducing LPIPS by 14.0%. These margins widen considerably against WPE and IRASim, where we see PSNR improvements of 4.09 dB and 3.06 dB and LPIPS reductions of 46.6% and 40.2%, respectively. The most pronounced gains occur on the wrist camera (SSIM +9.1%, PSNR +1.59 dB). This indicates that our closed-loop-aware post-training specifically excels at capturing the fine-grained object contact and hand-eye coordination details critical for downstream policy evaluation. Qualitatively, Fig. 3 confirms these improvements, while Fig. 4 illustrates a consistent distribution shift toward higher-fidelity generations. A 1-to-1 paired comparison reveals that our model outperforms the baseline in $\sim 98\%$ of validation samples, demonstrating the consistency of our gains across the entire dataset. Finally, Fig. 5 analyzes performance over extended rollouts; while both models naturally degrade as the horizon increases, our method maintains significantly higher fidelity over time.

Table 1: Visual quality metrics for 14-step autoregressive rollouts (≈ 11 s) on the DROID validation split. Values represent averages over the full rollout duration. Results marked with * are from [9]; † indicates our reproduction.

Evaluated Cameras	Model	Pixel/Structure		Perceptual
		SSIM $\uparrow$	PSNR $\uparrow$	LPIPS $\downarrow$
External Camera	WPE*	0.77	20.33	0.131
	IRASim*	0.77	21.36	0.117
	Ctrl-World*	0.83	23.56	0.091
	Ctrl-World†	0.84	23.02	0.081
	Ours	0.86	24.42	0.070
Wrist Camera	Ctrl-World†	0.62	17.80	0.310
	Ours	0.67	19.39	0.277

Comparison to Self-Forcing style baseline. Recent works such as Self-Forcing [18, 25] are not directly applicable to our case since they primarily distill from a (usually larger) bidirectional teacher, which is not available in our case. However, to measure the importance of our RL objective over the training on corrupted history, we post-trained the model on its own rollout history but ask it to regress the ground truth latents. On the validation set with rollout length of 10 interactions, the visual metrics favor ours: wrist **0.721/20.40/0.338** vs SF 0.710/20.17/0.338; external **Ours 0.872/25.52/0.116** vs 0.868/25.35/0.119.

Object- and Robot-Centric Evaluation. General-purpose world models often achieve high full-frame scores by over-optimizing for static background preservation while failing to capture the complex dynamics of manipulated objects. To evaluate task-relevant fidelity, we isolate the foreground using RoboEngine [40] to segment interacting objects and the robot arm. Computing metrics on these masked regions provides a rigorous measure of spatial and control consistency, which is more critical for downstream policy learning than raw background reconstruction. Table 2 confirms that our model’s gains are concentrated on these task-critical regions. On object-masked pixels, our improvements are even more pronounced than full-frame results: external camera LPIPS drops by 16.3% (vs. 14.0% full-frame), while wrist-camera SSIM improves by 4.1%. We observe similar trends for robot-centric metrics, with PSNR increasing by 1.63 dB and 1.74 dB for external and wrist views, respectively. These results demonstrate that our training objective successfully captures the intricate dynamics of robot-object interactions rather than relying on incidental background fidelity.

Human Preference Study. To complement automated evaluation, we conducted a blind human preference study to assess perceived realism and temporal consistency. Raters were presented with side-by-side video pairs from our model and the baseline, alongside the ground-truth video as a reference. They were tasked with selecting the rollout that appeared most realistic and remained most consistent with the ground-truth dynamics. Our model significantly outperforms the baseline, achieving an 80% preference rate (174 wins vs. 43). This substantial

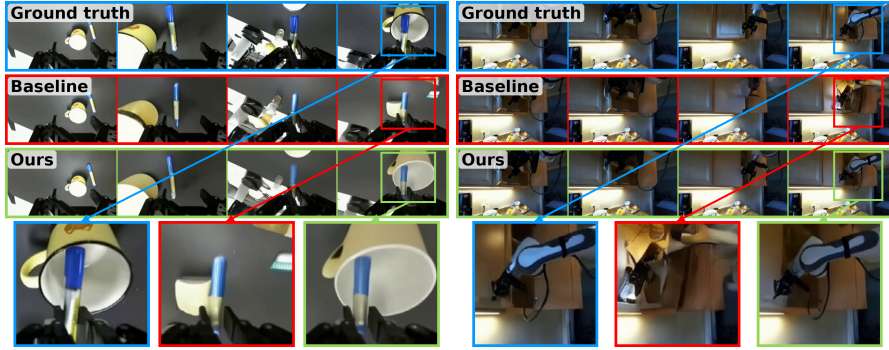


Fig. 3: Qualitative comparison of autoregressive rollout stability. We compare long-horizon (11 s) generations from the **baseline** [9] against our **PersistWorld** for the wrist camera. **Left: Object-centric fidelity.** The baseline model suffers from rapid decoherence; as errors compound in the history buffer, manipulated objects like the cup lose their structural identity and dissolve into amorphous textures. In contrast, our method maintains the spatial consistency and structural integrity of the object throughout the rollout. **Right: Robot-centric consistency.** The baseline exhibits significant **robot decoherence**, where the generated robot arm loses their geometric structure. Our approach maintains structural persistence. **Please see additional video results on the associated project page.**

margin is further reflected in the Elo ratings, where our model reaches 884.8 compared to the baseline’s 715.2.

These results confirm that our quantitative gains translate to a qualitatively superior experience, with human observers consistently favoring our model’s ability to maintain coherent dynamics over long horizons when compared directly against the ground-truth reference.



Fig. 6: Human preference results.

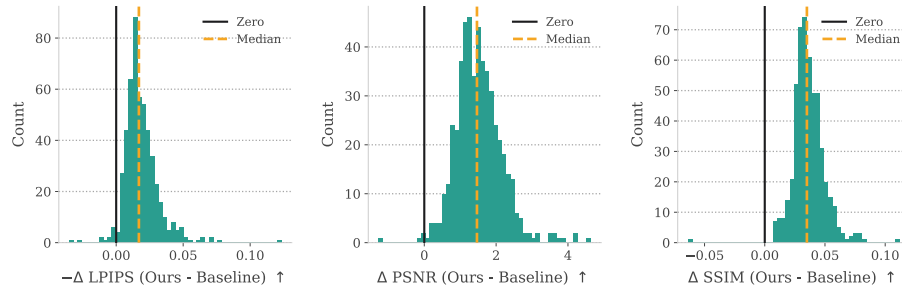


Fig. 4: Δ_{metric} of paired videos from the validation dataset. On 1–1 paired comparison, our PersistWorld world model is better than the baseline on $\sim 98\%$ of the sample ($p < 10^{-6}$).

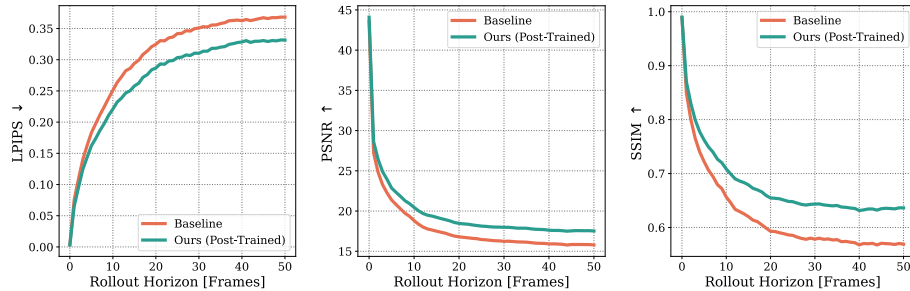


Fig. 5: Temporal evolution of wrist camera metrics. While both models exhibit natural degradation over longer horizons (x-axis), our post-trained model, PersistWorld (green), consistently maintains higher fidelity and slower error accumulation compared to the baseline (orange). Specifically, our method preserves a higher PSNR and SSIM while suppressing LPIPS drift, effectively extending the stable prediction horizon for complex, fine-grained interactions. See Fig. 11 in **Appendix. E.2** for external camera results.

Table 2: Masked visual metrics for 14-step autoregressive rollouts (≈ 11 s). We isolate object-only and robot-only pixels to evaluate task-relevant spatial and control consistency. Our model demonstrates superior fidelity in these dynamic regions compared to baselines, confirming that performance gains are driven by accurate interaction modeling rather than background reconstruction. All metrics are averaged over the full rollout horizon.

Evaluated Cameras	Model	Object-Only			Robot-Only		
		SSIM $\uparrow$	PSNR $\uparrow$	LPIPS $\downarrow$	SSIM $\uparrow$	PSNR $\uparrow$	LPIPS $\downarrow$
External Camera	Ctrl-World [†]	0.88	22.25	0.025	0.82	17.62	0.039
	Ours	0.89	23.60	0.021	0.86	19.25	0.033
Wrist Camera	Ctrl-World [†]	0.73	18.52	0.088	0.83	25.50	0.027
	Ours	0.76	19.87	0.078	0.86	27.24	0.023

Generalization beyond DROID. We finetuned the SVD backbone on Bridge (WidowX) [7] and RT-1 (Google Robot) [5] for 100k steps, then applied PersistWorld post-training. PersistWorld post-training improves all three metrics on both embodiments over the SVD backbone. On Bridge, SSIM $0.864 \rightarrow 0.876$, PSNR $25.20 \rightarrow 26.36$, LPIPS $0.1061 \rightarrow 0.0957$; on RT-1, SSIM $0.833 \rightarrow 0.865$, PSNR $23.81 \rightarrow 26.63$, LPIPS $0.1683 \rightarrow 0.1305$.

Policy Evaluation. Along with the visual metrics, we also evaluate the correlation between the world-model-based policy evaluations and the real-world policy evaluations. Across three tasks and three policies we find that PersistWorld has better correlation (0.822 vs 0.796) and MMRV (0.006 vs 0.053) than Ctrl-World. See Appendix C for more details.

Policy Improvement. We ran a Ctrl-World-style policy improvement experiment: π_0 is full-finetuned on 50 rollouts for 4 block spatial picking task from each WM, then evaluated on a real-robot (5 trials/block). Prompt: “Pick up the top-

left/top-right/bottom-left/bottom-right block". Average task progression: base 0.350, +Ctrl-World 0.558, +**Ours 0.667 (+19% over Ctrl-World)**.

5 Conclusion

In this paper, we addressed the critical challenge of exposure bias in action-conditioned robot world models. While existing diffusion-based world models produce high-fidelity short-term clips, their utility as simulators has been limited by compounding errors during autoregressive deployment. We introduced a reinforcement learning post-training framework that bridges this "closed-loop gap" by training the model on its own generated rollouts rather than ground-truth teacher forcing.

Our technical contributions—adapting a contrastive online reinforcement learning objective to x_0 -prediction backbones and designing a variable-length branching training protocol with multi-view perceptual rewards—enable the model to remain stable over longer horizons. Our approach establishes a new state-of-the-art on the DROID dataset, significantly reducing perceptual drift and maintaining structural integrity. The 80% preference rate in our human study and the marked improvement in object and robot-centric metrics suggest that RL post-training is a powerful tool for transforming video generators into reliable, persistent robot simulators. By stabilizing multi-step rollouts, this work paves the way for using world models as scalable, high-fidelity virtual environments for the evaluation and improvement of general-purpose robotic policies.

Limitations and Future Work. Despite these gains, our approach has limitations. The group-relative training protocol requires sampling $K = 16$ independent candidates per update, which increases computational overhead during post-training compared to standard supervised fine-tuning. However, a significant advantage of our framework is its modularity; the contrastive objective is reward-agnostic, meaning the model can be optimized against any combination of perceptual, physical, or task-specific signals. While our current implementation utilizes visual fidelity rewards (LPIPS, SSIM, PSNR) to stabilize the rollouts, these do not yet explicitly enforce physical or geometrical constraints. Future work will leverage this flexibility to explore more consistency rewards—including physics-informed constraints and geometry-aware metrics—to further enhance physical realism. Additionally, we intend to investigate the application of these persistent world models directly within the policy optimization loop.

Acknowledgements

This work was supported by the European Union’s Horizon Europe projects AGIMUS (No. 101070165), euROBIN (No. 101070596), ERC FRONTIER (No. 101097822), ELIAS (No. 101120237) and ELLIOT (No. 101214398). Compute resources and infrastructure were supported by the Ministry of Education, Youth and Sports of the Czech Republic through the e-INFRA CZ (ID:90254) and by the European Union’s Horizon Europe project CLARA (No. 101136607).

References

1. Bengio, S., Vinyals, O., Jaitly, N., Shazeer, N.: Scheduled sampling for sequence prediction with recurrent neural networks. *Advances in neural information processing systems* **28** (2015)
2. Black, K., Janner, M., Du, Y., Kostrikov, I., Levine, S.: Training diffusion models with reinforcement learning. *arXiv preprint arXiv:2305.13301* (2023)
3. Black, K., Nakamoto, M., Atreya, P., Walke, H., Finn, C., Kumar, A., Levine, S.: Zero-shot robotic manipulation with pretrained image-editing diffusion models. *arXiv preprint arXiv:2310.10639* (2023)
4. Blattmann, A., Dockhorn, T., Kulal, S., Mendelevitch, D., Kilian, M., Lorenz, D., Levi, Y., English, Z., Voleti, V., Letts, A., et al.: Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127* (2023)
5. Brohan, A., Brown, N., Carbajal, J., Chebotar, Y., Dabis, J., Finn, C., Gopalakrishnan, K., Hausman, K., Herzog, A., Hsu, J., et al.: Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817* (2022)
6. Du, Y., Yang, S., Dai, B., Dai, H., Nachum, O., Tenenbaum, J., Schuurmans, D., Abbeel, P.: Learning universal policies via text-guided video generation. *Advances in neural information processing systems* **36**, 9156–9172 (2023)
7. Ebert, F., Yang, Y., Schmeckpeper, K., Bucher, B., Georgakis, G., Daniilidis, K., Finn, C., Levine, S.: Bridge data: Boosting generalization of robotic skills with cross-domain datasets. *arXiv preprint arXiv:2109.13396* (2021)
8. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., Podell, D., Dockhorn, T., English, Z., Lacey, K., Goodwin, A., Marek, Y., Rombach, R.: Scaling rectified flow transformers for high-resolution image synthesis (2024), <https://arxiv.org/abs/2403.03206>
9. Guo, Y., Shi, L.X., Chen, J., Finn, C.: Ctrl-world: A controllable generative world model for robot manipulation. *arXiv preprint arXiv:2510.10125* (2025)
10. Ha, D., Schmidhuber, J.: World models. *arXiv preprint arXiv:1803.10122* **2**(3), 440 (2018)
11. Hafner, D., Lillicrap, T., Ba, J., Norouzi, M.: Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603* (2019)
12. Hafner, D., Lillicrap, T., Fischer, I., Villegas, R., Ha, D., Lee, H., Davidson, J.: Learning latent dynamics for planning from pixels. In: *International conference on machine learning*. pp. 2555–2565. PMLR (2019)
13. Hafner, D., Pasukonis, J., Ba, J., Lillicrap, T.: Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104* (2023)
14. Ho, J., Chan, W., Saharia, C., Whang, J., Gao, R., Gritsenko, A., Kingma, D.P., Poole, B., Norouzi, M., Fleet, D.J., et al.: Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303* (2022)
15. Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., Fleet, D.J.: Video diffusion models. *Advances in neural information processing systems* **35**, 8633–8646 (2022)
16. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: Lora: Low-rank adaptation of large language models (2021), <https://arxiv.org/abs/2106.09685>
17. Hu, Y., Guo, Y., Wang, P., Chen, X., Wang, Y.J., Zhang, J., Sreenath, K., Lu, C., Chen, J.: Video prediction policy: A generalist robot policy with predictive visual representations (2025), <https://arxiv.org/abs/2412.14803>

18. Huang, X., Li, Z., He, G., Zhou, M., Shechtman, E.: Self forcing: Bridging the train-test gap in autoregressive video diffusion. *Advances in Neural Information Processing Systems* **38**, 167283–167308 (2026)
19. Huynh-Thu, Q., Ghanbari, M.: Scope of validity of psnr in image/video quality assessment. *Electronics letters* **44**(13), 800–801 (2008)
20. Jordan, K., Jin, Y., Boza, V., You, J., Cesista, F., Newhouse, L., Bernstein, J.: Muon: An optimizer for hidden layers in neural networks (2024), <https://kellerjordan.github.io/posts/muon/>, Accessed: 2026-03-05
21. Karras, T., Aittala, M., Aila, T., Laine, S.: Elucidating the design space of diffusion-based generative models (2022), <https://arxiv.org/abs/2206.00364>
22. Khazatsky, A., Pertsch, K., Nair, S., Balakrishna, A., Dasari, S., Karamcheti, S., Nasiriany, S., Srirama, M.K., Chen, L.Y., Ellis, K., et al.: Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945* (2024)
23. Ko, P.C., Mao, J., Du, Y., Sun, S.H., Tenenbaum, J.B.: Learning to Act from Actionless Videos through Dense Correspondences. *arXiv:2310.08576* (2023)
24. Liu, J., Liu, G., Liang, J., Li, Y., Liu, J., Wang, X., Wan, P., Zhang, D., Ouyang, W.: Flow-grpo: Training flow matching models via online rl. *arXiv preprint arXiv:2505.05470* (2025)
25. Liu, K., Hu, W., Xu, J., Shan, Y., Lu, S.: Rolling forcing: Autoregressive long video diffusion in real time. *arXiv preprint arXiv:2509.25161* (2025)
26. Ma, Y., Wu, X., Sun, K., Li, H.: Hpsv3: Towards wide-spectrum human preference score. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 15086–15095 (2025)
27. Ning, M., Sangineto, E., Porrello, A., Calderara, S., Cucchiara, R.: Input perturbation reduces exposure bias in diffusion models. *arXiv preprint arXiv:2301.11706* (2023)
28. O’Neill, A., Rehman, A., Maddukuri, A., Gupta, A., Padalkar, A., Lee, A., Pooley, A., Gupta, A., Mandlekar, A., Jain, A., et al.: Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In: *2024 IEEE International Conference on Robotics and Automation (ICRA)*. pp. 6892–6903. IEEE (2024)
29. Prabhudesai, M., Mendonca, R., Qin, Z., Fragkiadaki, K., Pathak, D.: Vader: Video diffusion alignment via reward gradients
30. Quevedo, J., Sharma, A.K., Sun, Y., Suryavanshi, V., Liang, P., Yang, S.: Worldgym: World model as an environment for policy evaluation. *arXiv preprint arXiv:2506.00613* (2025)
31. Ranzato, M., Chopra, S., Auli, M., Zaremba, W.: Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732* (2015)
32. Ren, A.Z., Lidard, J., Ankile, L.L., Simeonov, A., Agrawal, P., Majumdar, A., Burchfiel, B., Dai, H., Simchowitz, M.: Diffusion policy policy optimization. *arXiv preprint arXiv:2409.00588* (2024)
33. Rigter, M., Gupta, T., Hilmkil, A., Ma, C.: Avid: Adapting video diffusion models to world models. *arXiv preprint arXiv:2410.12822* (2024)
34. Singer, U., Polyak, A., Hayes, T., Yin, X., An, J., Zhang, S., Hu, Q., Yang, H., Ashual, O., Gafni, O., et al.: Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792* (2022)
35. Wallace, B., Dang, M., Rafailov, R., Zhou, L., Lou, A., Purushwalkam, S., Ermon, S., Xiong, C., Joty, S., Naik, N.: Diffusion model alignment using direct preference optimization. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 8228–8238 (2024)

36. Wang, Z., Wang, T., Zhang, H., Zuo, X., Wu, J., Wang, H., Sun, W., Wang, Z., Cao, C., Zhao, H., et al.: Worldcompass: Reinforcement learning for long-horizon world models. arXiv preprint arXiv:2602.09022 (2026)
37. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* **13**(4), 600–612 (2004)
38. Wu, J., Yin, S., Feng, N., Long, M.: Rlvr-world: Training world models with reinforcement learning. arXiv preprint arXiv:2505.13934 (2025)
39. Xue, Z., Wu, J., Gao, Y., Kong, F., Zhu, L., Chen, M., Liu, Z., Liu, W., Guo, Q., Huang, W., et al.: Dancegrpo: Unleashing grpo on visual generation. arXiv preprint arXiv:2505.07818 (2025)
40. Yuan, C., Joshi, S., Zhu, S., Su, H., Zhao, H., Gao, Y.: Roboengine: Plug-and-play robot data augmentation with semantic robot segmentation and background generation (2025), <https://arxiv.org/abs/2503.18738>
41. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 586–595 (2018)
42. Zheng, K., Chen, H., Ye, H., Wang, H., Zhang, Q., Jiang, K., Su, H., Ermon, S., Zhu, J., Liu, M.Y.: Diffusionnft: Online diffusion reinforcement with forward process. arXiv preprint arXiv:2509.16117 (2025)
43. Zhu, C., Yu, R., Feng, S., Burchfiel, B., Shah, P., Gupta, A.: Unified world models: Coupling video and action diffusion for pretraining on large robotic datasets. arXiv preprint arXiv:2504.02792 (2025)
44. Zhu, F., Wu, H., Guo, S., Liu, Y., Cheang, C., Kong, T.: Irasim: A fine-grained world model for robot manipulation. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 9834–9844 (2025)