CTRL-WORLD: A CONTROLLABLE GENERATIVE WORLD MODEL FOR ROBOT MANIPULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Generalist robot policies can now perform a wide range of manipulation skills, but evaluating and improving their ability with unfamiliar objects and instructions remains a significant challenge. Rigorous evaluation requires a large number of real-world rollouts, while systematic improvement demands additional corrective data with expert labels. Both of these processes are slow, costly, and difficult to scale. World models offer a promising, scalable alternative by enabling policies to rollout within imagination space. However, a key challenge is building a controllable world model that can handle multi-step interactions with generalist robot policies. This requires a world model compatible with modern generalist policies by supporting multi-view prediction, fine-grained action control, and consistent long-horizon interactions, which is not achieved by previous works. In this paper, we make a step forward by introducing a controllable multi-view world model that can be used to evaluate and improve the instruction-following ability of generalist robot policies. Our model maintains long-horizon consistency with a pose-conditioned memory retrieval mechanism and achieves precise action control through frame-level action conditioning. Trained on the DROID dataset (95k trajectories, 564 scenes), our model generates spatially and temporally consistent trajectories under novel scenarios and new camera placements for over 20 seconds. We show that our method can accurately rank policy performance without real-world robot rollouts. Moreover, by synthesizing successful trajectories in imagination and using them for supervised fine-tuning, our approach can improve policy success by 44.7%. Videos can be found at `https://sites.google.com/view/ctrl-world`.

1 INTRODUCTION

Recent advances in vision-language-action (VLA) models have demonstrated competence across a wide range of manipulation tasks and scenarios (Black et al., 2024; Intelligence et al., 2025; Pertsch et al., 2025; Wen et al., 2025; Brohan et al., 2023; Kim et al., 2024; Cui et al., 2025). Despite their promise, current policies remain brittle when tested in open-world circumstances. A central challenge is *policy evaluation*. Assessing generalist policy performance typically requires large numbers of real-world rollouts, carefully repeated across tasks and environments to achieve statistical significance (Atreya et al., 2025). Such protocols are logistically demanding, slow down iteration, and inhibit nuanced understanding of current policy capabilities. Equally critical is *policy improvement*: once weaknesses are revealed, existing methods offer few ways to strengthen policies on failure cases beyond collecting more expert data. Although large-scale pretraining provides some robustness, policies often remain fragile when they encounter unfamiliar objects or instructions. What is missing is a fast and cheap feedback-driven mechanism for refining generalist models: a way to surface failure cases, gather corrective experiences, and iteratively improve the policy.

Learning a predictive model and iterating in imagination is a scalable and promising alternative. While prior work has explored action-conditioned world models, most approaches focus on passive video prediction settings and are not sufficient to actively interact with advanced generalist policies (Li et al., 2025b; Zhu et al., 2024). We observe several important limitations that hinder their ability to support policy-in-the-loop rollouts. First, these models typically simulate only a single third-person camera view, which can lead to severe partial observability and, in turn, cause hallucinations (e.g., an object snapping into the gripper without prior physical contact). This single-view input is also incompatible with many modern VLA policies that require both third-person and wrist-view cameras as input. Moreover, existing models typically lack the fine-grained control required to capture the

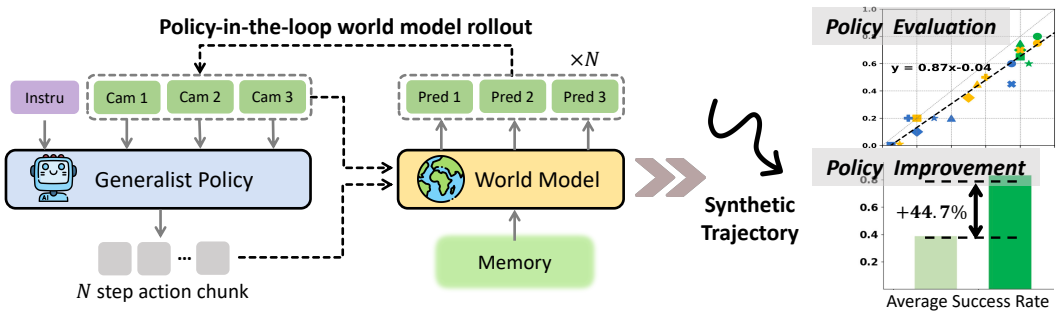

Figure 1: Ctrl-World is designed for *policy-in-the-loop* rollouts with generalist robot policies. It generates joint multi-view predictions (including wrist views), enforces fine-grained action control via frame-level conditioning, and sustains coherent long-horizon dynamics through pose-conditioned memory retrieval. These components enable (1) accurate policy *evaluation* in imagination, with alignment to real-world rollouts, and (2) targeted policy *improvement* through synthetic trajectories.

causal effects of high-frequency actions. Finally, they struggle to maintain temporal consistency across long-horizon video generations.

In this paper, we introduce **Ctrl-World**, a **Controllable**, multi-view generative **world** model designed for policy-in-the-loop interaction, enabling multi-step rollouts entirely within imagination space, as illustrated in Figure 1. Our design relies on three key components: (1) Joint multi-view prediction captures a more comprehensive visual representation of the scene and meets the input format of modern VLA policies. Notably, the inclusion of wrist-camera prediction significantly reduces hallucinations during contact-rich object interactions. (2) Frame-level action conditioning tightly aligns visual dynamics with control signals, ensuring that generated rollouts reflect the causal effect of each action. (3) Memory retrieval, which adds sparse history frames into the context and projects corresponding pose information into each frame, allows the model to attend to similar past states and retrieve relevant information. This mechanism stabilizes long-horizon rollouts and preserves temporal consistency. Together, these mechanisms allow us to transform a pre-trained passive video generator into a policy-compatible interactive simulator.

The core contribution of this work is a *controllable world model* for robot manipulation. In experiments, we find this model enables a new imagination-based workflow in which policies can be both *evaluated*—with ranking alignment to real-world rollouts—and *improved*—through targeted synthetic data that boosts success rates. Specifically, we train Ctrl-World on the DROID dataset (Khazatsky et al., 2024) and show that it generalizes to novel scenes and camera placements, sustaining coherent rollouts for over 20 seconds. We further show that imagination-based evaluations with Ctrl-World faithfully reflect policies' real-world instruction-following ability. Finally, we demonstrate that we can improve the performance of $\pi_{0.5}$-DROID (Intelligence et al., 2025) on downstream tasks with unseen objects and novel instructions by synthesizing successful trajectories inside the world model and performing supervised fine-tuning with these synthetic roll-outs.

2 Related Works

Video Generation Models for Robotics. Recent advances in video generation models (Agarwal et al., 2025; Wan et al., 2025; Blattmann et al., 2023a) have enabled the creation of realistic and temporally consistent content, reflecting a strong understanding of the physical world. Some works leverage video prediction models to synthesize robotic trajectories with fake action labels, and these synthetic trajectories can then be used for policy learning (Jang et al., 2025; Bharadhwaj et al., 2024). Other works directly employ video models as policy backbones, decoding actions through tracking or inverse dynamics (Black et al., 2023; Du et al., 2024; Yang et al., 2023; Hu et al., 2024; Liang et al., 2024; Liao et al., 2025; Tan et al., 2025; Feng et al., 2025). A complementary line of research integrates future-prediction objectives into generalist policies via co-training (Zhao et al., 2025; Li et al., 2025a; Zhu et al., 2025; Guo et al., 2024; Gao et al., 2024; Zhang et al., 2025; Zheng et al., 2025; Zhong et al., 2025), incorporating physical knowledge into the policy. Unlike these works, we leverage video generation to perform action-conditioned prediction, which enables using the model for both policy evaluation and policy improvement.

Action-Conditioned World Models. Although pretrained video models are powerful, they are often only conditioned on high-level language instructions. Nonetheless, some prior works have explored using action-conditioned predictive models, both in low-dimensional state spaces (Nagabandi et al., 2020) and with image observations Hafner et al. (2019; 2020); Hansen et al. (2022); Wu et al. (2023); Oh et al. (2015). Many of these approaches learn task-specific models (Hafner et al., 2019), while we focus on training generalist, multi-task world models. Building on early works (Finn & Levine, 2017; Ebert et al., 2018; Xie et al., 2019; Dasari et al., 2019; Yang et al., 2023; Wu et al., 2024) as well as more recent approaches that leverage diffusion (Quevedo et al., 2025; Chen et al., 2024; Ball et al., 2025; Gao et al., 2025) and frame-level action conditioning (Zhu et al., 2024), we propose a model that incorporates multi-view prediction, long-horizon temporal coherence, and fine-grained controllability. Our experiments show that these capabilities enable effective evaluation and improvement of state-of-the-art generalist VLA policies.

3 Problem Formulation

We aim to develop a world model that can predict the future outcomes of actions proposed by a generalist robot policy. A modern generalist policy π typically maps multi-view observations and language instructions into a sequence of actions (Zhao et al., 2023; Black et al., 2025). Specifically, robot observation $o_t = [I_t^1, \dots, I_t^n, q_t]$ includes n camera views $[I_t^1, \dots, I_t^n]$ and robot pose q_t, the policy outputs an H-step action chunk given an instruction l:

$$a_{t+1}, a_{t+2}, ..., a_{t+H} \sim \pi(\cdot|o_t, l) \tag{1}$$

Our goal is to use a world model W to predict the outcomes of executing each step in $A_t = [a_{t+1}, \dots, a_{t+H}]$. To enable multi-step interaction with the policy in imagination space, W must generate future multi-view observations:

$$o_{t+1}, ..., o_{t+H} \sim W(\cdot|o_t, A_t) \tag{2}$$

Then the final prediction o_{t+H} can be send back to policy π to produce the next action chunk $A_{t+H} \sim \pi(\cdot|o_{t+H}, l)$. In this way, the policy and world model interact auto-regressively, enabling long-horizon rollouts entirely within imagination space.

4 Controllable World Model for Robot Manipulation

4.1 Learning World Model Ctrl-World

Our goal is to learn a world model that can be used to evaluate and improve modern VLA policies. To achieve this, the model must first support multiview observations that are commonly used by such policies. It is also important for the model to be controllable — reliably and closely follow the action inputs — even when initialized from a pre-trained backbone that lacks such control. Finally, the model must maintain temporal consistency over long horizons, even in the presence of occlusions, to produce coherent rollouts. We initialize our world model from a pretrained video diffusion backbone with spatial-temporal transformers (Blattmann et al., 2023b) and introduce three key adaptations, illustrated in Figure 2.

Multi-View Joint Predictions. State-of-the-art VLA models often rely on multiple third-person cameras for global context and wrist-mounted cameras for precise interactions (Intelligence et al., 2025; Liu et al., 2024; 2025). To match this, the world model must generate spatially consistent predictions across all views at each step. Prior work has shown that feed-forward transformers can effectively capture spatial relationships between multi-view cameras in a scalable manner (Wang et al., 2025). Following prior work, we concatenate the N input images—each containing $H \times W$ tokens—along the token dimension and jointly predict all views $o_{t:t+H}$. In experiments, we find multi-view joint prediction also improves consistency and substantially reduces hallucinations.

Pose-conditioned Memory Retrieval Mechanism. Prediction errors in world models tend to accumulate over long rollouts, leading to drift and incoherence. To mitigate this, we augment the model input with past frames. To prevent the context from becoming too long, we sample k history frames with a stride m, enabling the model to predict $o_{t+1:t+H} \sim W(\cdot|o_{t-km}, ..., o_t, l)$. Additionally,

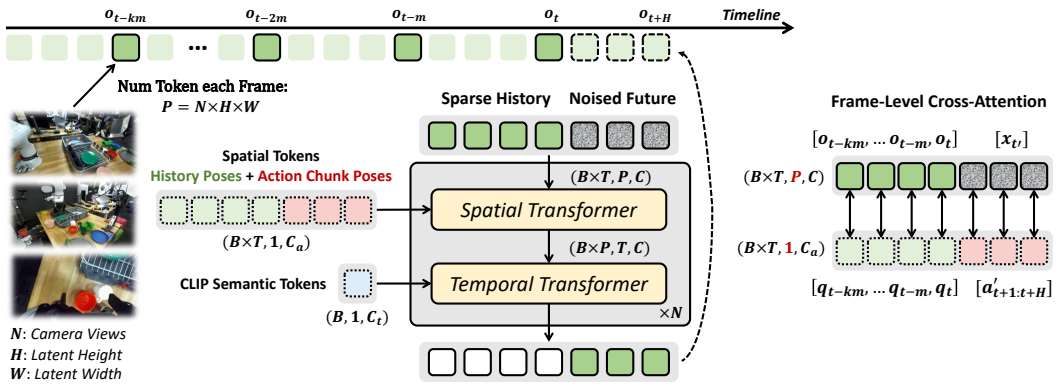

Figure 2: Ctrl-World is initialized from a pretrained video diffusion model and adapted into a controllable, temporally consistent world model with: (1) Multi-view input and joint prediction for unified information understanding. (2) Memory retrieval mechanism, which adds sparse history frames in context and project pose information into each frame via frame-level cross-attention, re-anchoring predictions to similar past states. (3) Frame-level action conditioning to better align high-frequency action with visual dynamics.

we embed the corresponding robot arm poses $[q_{t-km}, ..., q_t]$ into frames $[o_{t-km}, ..., o_t]$ via frame-wise cross-attention within spatial transformer. This allows the model to use the arm pose to identify relevant frames from the past, effectively anchoring future predictions to relevant history.

Frame-level Action Conditioning. The pretrained video model conditions only on text and image, which limits its control precision. To enable full controllability, we additionally condition the model on the action sequence $[a_{t+1:t+H}]$ output by the policy. We also transform each action sequence into Cartesian-space robot arm poses $[a'_{t+1:t+H}]$ and concatenate with past poses $[q_{t-km}, ..., q_{t-m}, q_t]$. Frame-wise cross-attention is then applied within the spatial transformer, allowing the visual tokens of each frame to attend to its associated pose embedding. For history frames, this pose corresponds to $[q_{t-km}, ..., q_{t-m}, q_t]$, while for future frames, it corresponds to $[a'_{t+1:t+H}]$.

Training Objective. We initialize our model with the pretrained 1.5B Stable-Video-Diffusion (SVD) model (Blattmann et al., 2023a). To inherit the knowledge and structure in the pretrained video model, we only newly initialize an action-projection MLP for the input actions and keep other parameters unchanged at initialization. Then this action-conditioned world model is fine-tuned with diffusion loss (Ho et al., 2020; Karras et al., 2022). During training, the prediction target $x_0 = o_{t+1:t+H}$ is perturbed with Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$ at diffusion step $t' \in [0, T']$ with scheduler $\overline{\alpha_{t'}}$, resulting in $x_{t'} = \sqrt{\overline{\alpha_{t'}}} x_0 + \sqrt{1 - \overline{\alpha_{t'}}} \epsilon_{t'}$. The model input is the concatenation of history tokens and the noised future: $[o_{t-km}, ..., o_{t-m}, o_t, x_{t'}]$. The overall training objective is:

$$\mathcal{L} = \mathbb{E}_{x_0, \epsilon, t'} \| \hat{x}_0(x_{t'}, t', c) - x_0 \|^2 \tag{3}$$

where $\hat{x}_0$ denotes the model's prediction, and $c = [q_{t-km}, ..., q_t, a'_{t+1:t+H}, o_{t-km}, ..., o_t]$ corresponds to all model inputs. More details of the model can be found in the Appendix A.

4.2 Using Ctrl-World for Policy Evaluation and Improvement

Policy Evaluation within World Model. Once a controllable and consistent world model is trained, we can conduct policy-in-the-loop rollouts in imagination space. Given an initial observation o_0 and instruction l, a policy π together with the world model W can generate a synthetic trajectory τ. The initial observation can be sampled from the validation dataset or recorded as a snapshot from a real-world setup. In our experiments, we label each trajectory as a success or failure based on human preference judgments. While recent works (Du et al., 2023) explore the use of Vision-Language Models as general-purpose reward models, we leave such extensions to future work.

Policy Improvement with Synthetic Data. Beyond evaluation, the world model enables searching for successful synthetic trajectories to improve policy performance. We observe that, under fixed initial observations and instructions, policy behavior tends to be highly deterministic. For example, the policy tends to grasp the same object across multiple trials, rather than stochastically reaching for

Algorithm 1 World Model Rollout and Policy Improvement

Given: policy π_θ, action perturbation function ϵ_a, world model W, task instructions $[l^0, ., l^M]$ with initial obs $[o_0^0, ., o_0^M]$, synthetic dataset D_s, interaction step N, action horizon H.

for $i = 0$ **to** M **do**
  $\tau = [o_0^i]$
  for $j = 0$ **to** N **do**
    Current observation: $o_t = \tau[t]$ where $t = j * H$
    Sample action from perturbed policy: $a_{t+1:t+H} = \pi_\theta(o_t, l, \epsilon_a)$ ▷ For diverse rollouts
    Prepare history context: $h = [o_{t-km}, ..., o_{t-2m}, o_{t-m}]$
    Make predictions with world model: $o_{t+1:t+H} = W(h, o_t, a_{t+1:t+H})$
    Add predictions into trajectory: $\tau = \tau \cup o_{t+1:t+H}$.
  end for
  Judge success of τ based on human-preference. Add τ into D_s if success.
end for
Finetune π_θ with $\mathcal{L}_\theta = \mathbb{E}_{o_t, a_{t:t+H} \sim D_s} \|\pi_\theta(o_t, l) - a_{t:t+H}\|^2$.

various objects. To explore a larger search space, we introduce structured perturbations to encourage diversity in rollouts. Specifically, we can (i) rephrase the instructions, since VLA policies tend to be steerable, exhibiting different behaviors in response to different instructions; or (ii) reset the policy to random initial states within the world model, which leads to diverse initial observations. Starting from a set of downstream tasks with language instructions $[l^0, \ldots, l^M]$, we collect synthetic rollouts and score them based on human preference. To improve the policy performance, we fine-tune the policy on successful trajectories. The overall procedure is summarized in Algorithm 1.

5 Experiments

In this section, we conduct experiments to evaluate Ctrl-World. We aim to answer the following questions: (1) Can Ctrl-World generate long-horizon rollouts that are both spatially and temporally consistent, while maintaining high controllability? (2) Can Ctrl-World reliably evaluate different generalist robot policies in imagination space, faithfully reproducing their real-world performance rankings? (3) Can Ctrl-World improve a policy's instruction following by discovering and synthesizing successful trajectories entirely within its imagination?

5.1 Experiment Setups

DROID Platform and Dataset. Our experiments use the DROID platform (Khazatsky et al., 2024), which features a Panda robot arm equipped with a Robotiq Gripper. The platform includes one wrist-view camera and two randomly positioned third-view cameras that observe the workspace. The DROID dataset (Khazatsky et al., 2024) contains 95,599 diverse trajectories collected from 564 scenes, providing dense coverage of the workspace. This includes about 76k successful and about 19k failed trajectories. The inclusion of diverse actions and failure data is crucial, as it allows us to train a controllable world model that can simulate a wide range of future scenarios.

Training Details. During training, our model jointly predicts outputs from all three cameras, each with a resolution of 192x320. The model is conditioned on a history of 7 frames, with an interval of 1-2 seconds between frames. We condition the model on the next 15 future actions, which corresponds to a one second action chunk in DROID. During interaction, if a policy's output is less than 15 steps, we pad the action chunk with dummy actions and only use the predictions for valid actions. We train the model on 2×8 H100 GPUs, with a total batch size of 64. Training takes approximately 2-3 days.

5.2 World Model Quality Analysis

Baselines and Evaluation Matrices. We quantitatively compare our model, Ctrl-World, against two prior action-conditioned world models: World-model-based Policy Evaluation (WPE) (Quevedo et al., 2025) and IRASim (Zhu et al., 2024). Since these models only predict from a single third-person camera view, we train a single-view version, Ctrl-World-third-view, which only inputs and predicts on

Evaluated Camera	Method	Computation-based		Model-based		
		PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓	FVD ↓
Third-view Camera	WPE	20.33	0.772	0.131	25.50	156.4
	IRASim	<u>21.36</u>	0.774	0.117	26.46	138.1
	Ctrl-World-Third-View	21.27	<u>0.793</u>	<u>0.110</u>	**23.47**	<u>127.5</u>
	Ctrl-World (ours)	**23.56**	**0.828**	**0.091**	<u>25.00</u>	**97.4**

Table 1: Quantitative results for interactive long-trajectory generation on the validation set. We evaluate our world model's quality by generating 10-second trajectories. Given a randomly sampled initial frame, the model receives a 15-step action chunk (spanning over 1 second) in each interaction and generates for 10 rounds auto-regressively. The results are averaged over 256 clips.

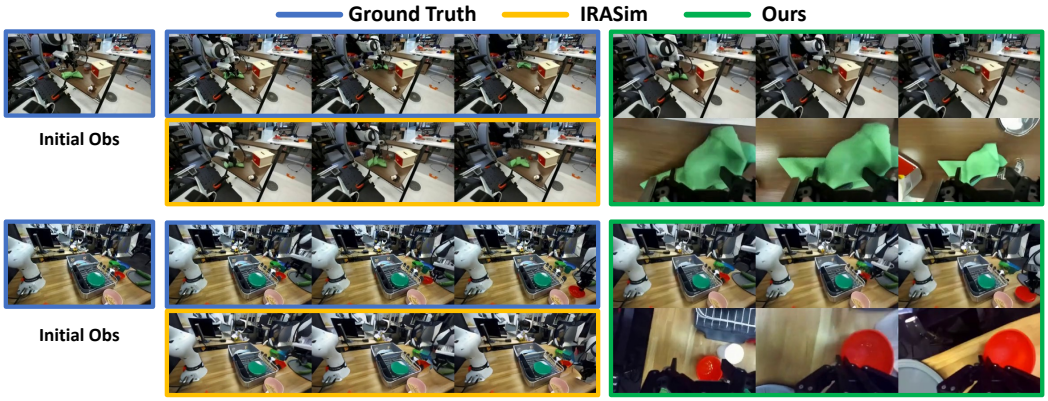

Figure 3: Qualitative results on long-horizon rollouts from the validation set. Prior models rely on single-view prediction, suffering from partial observability and hallucinations (e.g., failing to move the green towel or grasp the red bowl). In contrast, Ctrl-World jointly predicts from third-view and wrist-view cameras, yielding precise future trajectories aligned with the ground truth.

Evaluated Camera	Method	Computation-based		Model-based		
		PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓	FVD ↓
Third-view Camera	Ctrl-World	**23.56**	**0.828**	**0.091**	**25.00**	**97.4**
	Ctrl-World w/o memory	23.06	0.812	0.099	26.14	105.5
	Ctrl-World w/o frame-level cond	21.20	0.789	0.109	27.52	122.7
Wrist-view Camera	Ctrl-World	**19.18**	**0.665**	**0.252**	**25.78**	**127.1**
	Ctrl-World w/o memory	18.84	0.655	0.265	26.23	133.1
	Ctrl-World w/o frame-level cond	15.69	0.571	0.375	33.51	179.1
	Ctrl-World w/o joint pred	15.94	0.580	0.345	26.46	158.1

Table 2: Ablations on key components in Ctrl-World. Removing memory mechanisms, frame-level action conditioning or multi-view joint predictions all lead to a performance drop.

a single third-person camera for a fair comparison. For evaluation, we hold out 2% of the trajectories as a validation set and randomly sample 256 video clips, each 10 s in length. During rollouts, the world model receives 15-step action chunks (corresponding to 1 s) and autoregressively predicts the next frames for 10 steps, producing 10 s-long trajectories. We then compare the predicted videos against ground truth using both computational (PSNR (Hore & Ziou, 2010) and SSIM (Wang et al., 2004)) and model-based metrics (LPIPS (Zhang et al., 2018), FID (Heusel et al., 2017), and FVD (Unterthiner et al., 2018)).

Quantitative and Qualitative Results on Multi-step Interaction Trajectories. As shown in Table 1, Ctrl-World-third-view outperforms these prior models, and multi-view joint prediction further improves generation quality. Consistent with observations from prior work (Quevedo et al., 2025; Zhu et al., 2024), we also find that these baselines struggle to capture robot–object interactions and often generate hallucinated predictions. For instance, as shown in Figure 3, single-view prediction methods WPE, IRASim and Ctrl-World-third-view all fail to move the green towel or the red bowl. In contrast, Ctrl-World precisely models the robot–object interactions through joint prediction of the wrist-camera view, which provides critical, fine-grained information about contact events and object state changes.

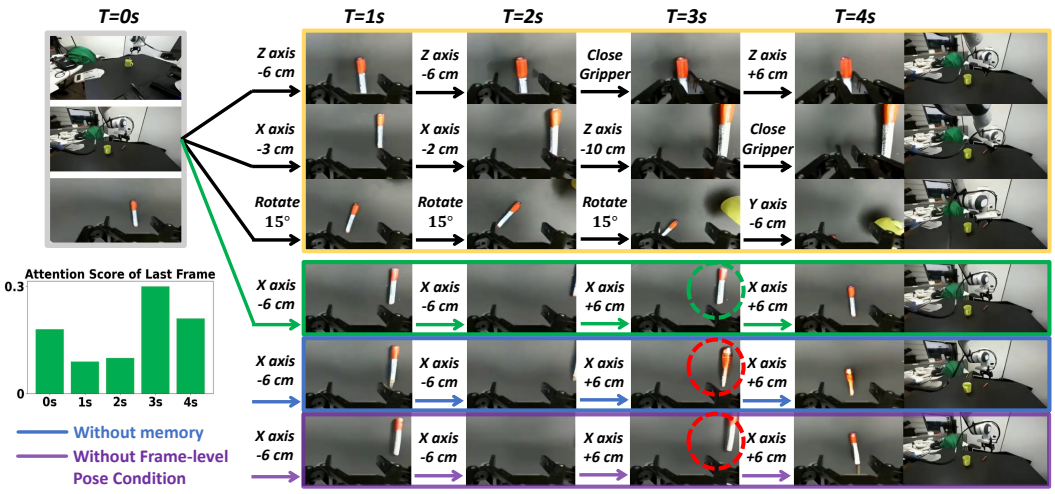

Figure 4: Controllability of Ctrl-World and ablations. Different action sequences can produce distinct rollouts in Ctrl-World with centimeter-level precision. Removing memory leads to blurry predictions (blue), while removing frame-level pose conditioning reduces control precision (purple). Attention visualization (left) when predicting the $t = 4$ s frame shows strong attention to the $t = 0$ s frame with the same pose, illustrating the effectiveness of memory retrieval. For clarity, each action chunk is expressed in natural language (e.g., "Z-axis -6 cm"). Due to space constraints, only the wrist-view is visualized for intermediate frames.

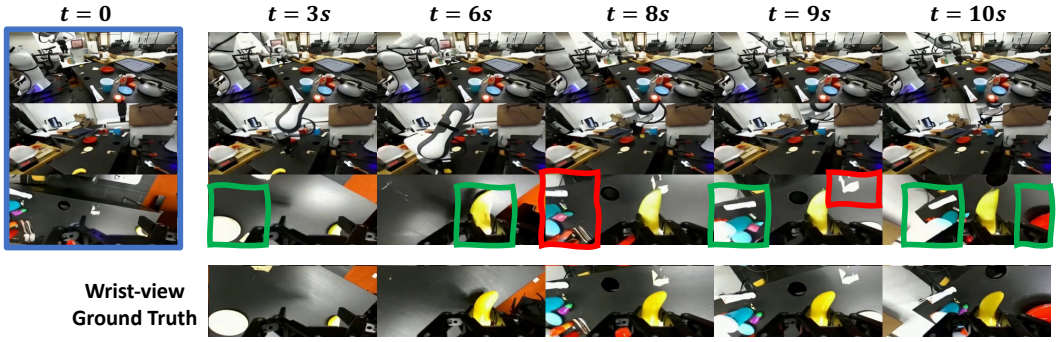

Figure 5: Consistency of Ctrl-World. Since the wrist camera's field of view changes dramatically within a single trajectory, leveraging multi-view information and memory retrieval is essential for generating consistent wrist-view predictions. Prediction highlighted in the green box are inferred from other camera views, while those in the red box are retrieved from memory.

Controllability of the World Model. A key requirement of a world model is the ability to simulate diverse future outcomes conditioned on different actions. We find that our model exhibits fine-grained controllability, producing precise future predictions even for actions that differ by only a few centimeters (see Figure 4). We hypothesize that this controllability arises from two main factors: first, the dense action space coverage in the DROID dataset; and second, our use of multi-view prediction and frame-level action conditioning, which is also supported by our ablation studies. On the left side of Figure 4, we visualize the attention weights when predicting the $t = 4$ s frame and observe strong attention to the $t = 0$ s frame with a similar pose, highlighting the effectiveness of our memory retrieval mechanism.

Consistency of the World Model. For the wrist camera, since the camera's field of view changes dramatically within a single trajectory, it is challenging for models to generate consistent, long-term predictions. As shown in Figure 5, we find that our model effectively leverages relevant information from both other camera views and historical frames, enabling it to generate consistent wrist-view predictions. Ablations on memory components and frame-level conditions are in Table 2, which confirm the importance of each component.

5.3 World Model for Policy Evaluation

In this section, we evaluate whether Ctrl-World can be used to evaluate the instruction-following ability of generalist robot policies and accurately reflect their performance rankings in the real world. We set up our own DROID platform and randomly place two third-person cameras around the workspace. Similar to how prior works have seen DROID policies generalize to new setups (Pertsch et al., 2025), we find that Ctrl-World, pretrained solely on the open-sourced DROID dataset, *can make accurate future predictions zero-shot in our newly configured scene with novel camera placements.*

Policies and Tasks. We evaluate three publicly released policies, π_0 (Black et al., 2023), π_0-FAST (Pertsch et al., 2025), and $\pi_{0.5}$ (Intelligence et al., 2025), across diverse tasks including Pick-and-Place, Towel-Folding, Drawer, Wipe-Table, Close-Laptop, Pull-tissue and Stack tasks on our DROID platform. We initialize real-world and world model rollouts with the same initial observations and execute each policy, following Algorithm 1. We report instruction following rates and success rates in Figure 7 and visualize qualitative comparisons between real and imagined rollouts in Figure 6. More rollout details can be found in Appendix B.

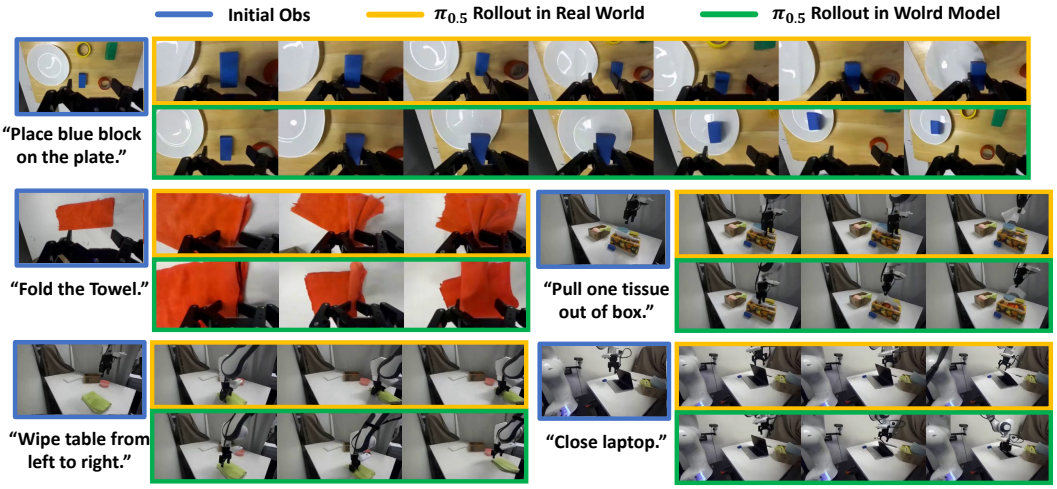

Figure 6: Comparisons between $\pi_{0.5}$ rollouts in the real-world and world model. Each trajectory contains 20 interactions between $\pi_{0.5}$ and Ctrl-World. Remarkably, both the generalist policy and Ctrl-World transfer zero-shot to our new DROID setup.

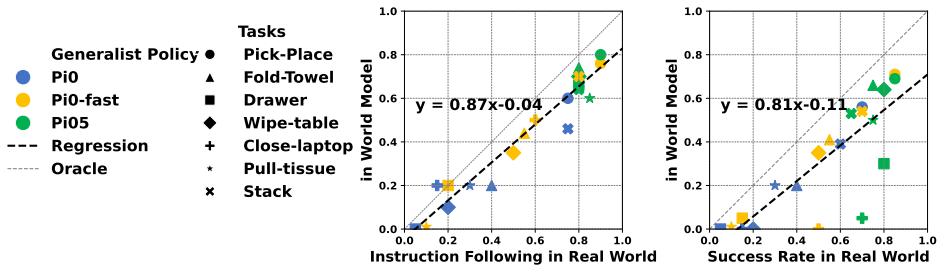

Figure 7: Quantitative correlations between real-world and world-model rollouts. The world model reliably captures instruction-following behavior but tends to underestimate the execution success rate.

Comparison Between Real-World and World Model Rollouts. Our results show that policy's high-level instruction-following behavior in the world model is closely correlated with that observed in the real world. However, we notice some gaps in evaluating low-level execution, specifically in precise modeling of complex physics dynamics such as collisions, objects sliding away, rotations, etc. (e.g., interaction with laptop is imprecise in Figure 6). We also observe that generalist policies tend to keep retrying in the real world after failed attempts, which the world model sometimes does not capture. Although some failure trajectories are included in the DROID dataset, there are still many failure modes outside the data distribution. We expect that collecting additional in-domain policy rollout data would improve the fidelity of the learned dynamics and narrow this gap (Team, 2025).

5.4 World Model for Policy Improvement

Post-train Policy with Synthetic Data. We now evaluate whether Ctrl-World can be used to generate synthetic post-training data for improving VLA models without real-world data. We use $\pi_{0.5}$ as our base policy and follow Algorithm 1. As described in Section 4.2, we encourage rollout diversity by either (1) rephrasing task instructions or (2) resetting the robot arm to a new initial state. For rephrasing, we call an LLM API (Team et al., 2023) to paraphrase instructions (e.g., transforming "place glove in box" into "pick up the cloth and put it inside the box"). For resetting, we randomly sample a new target initial position and move the robot arm there using a linear-interpolation motion planner before policy-interaction begins. We generate 400 trajectories per task and retain 25–50 successful trajectories based on human preference judgments. This selection step could be automated with reward models, which is an active area of research (Ma, 2025). Finally, we fine-tune the policy on the curated synthetic dataset for 2k steps, improving base model's capability in unfamiliar instructions and objects.

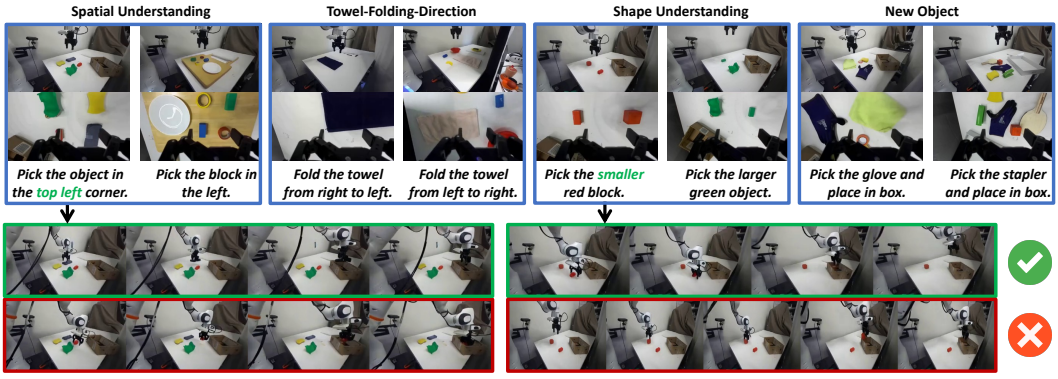

Figure 8: The top row illustrates examples of post-training tasks, while the bottom row presents synthetic trajectories generated within the world model. The world model can produce both successful and failed rollouts; we keep the successful trajectories and use them for policy fine-tuning.

Results. Some representative task examples and synthetic trajectories are visualized in Figure 8, and quantitative results are reported in Figure 9. While the pretrained $\pi_{0.5}$ policy achieves low success rates on unfamiliar objects and novel instructions, post-training aligns the model with new instructions and boosts the success rate from 38.7% to 83.4% on these downstream tasks. We include task details in Appendix C.

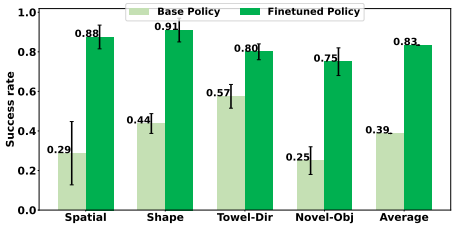

Figure 9: Policy improvement. Post-training on synthetic data improves policy instruction-following by 44.7% on average.

6 Conclusion

We presented Ctrl-World, a controllable world model for robot manipulation that supports closed-loop policy evaluation and improvement entirely within the model's imagination. Policies evaluated in Ctrl-World exhibit instruction-following behaviors that closely mirror those in the real world. Notably, post-training on generated data boosts the pretrained robot policy's success rate on novel instructions from 38.7% to 83.4%.

Despite these promising results, important challenges remain. Our model can fail on tasks involving precise interactions or long-horizon reasoning, and performance is sensitive to initial observations. These limitations may diminish as video backbones become more physically accurate and coherent over time (Ball et al., 2025; Agarwal et al., 2025). In addition, our experiments focus on improving instruction following, and we expect that our model is not accurate enough to improve performance in other aspects such as the low-level success rate on previously seen instructions. Improving the model with iterative policy roll-out and fine-tuning is an exciting future direction. Looking forward, we believe generative world models can transform how robots acquire new skills, enabling scalable policy evaluation and allowing them to learn not just from real world experience, but also safely and efficiently from generated experience.

7 Reproducibility statement

We include the code in the supplementary material. We will further clean up the code and make it publicly available.

8 Ethics statement

We are not aware of any significant ethical concerns related to this work at the current stage.

9 The Use of Large Language Models (LLMs)

We only used LLMs to rephrase and polish the text for clarity and readability.

References

Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, et al. Cosmos world foundation model platform for physical ai. *arXiv preprint arXiv:2501.03575*, 2025.

Pranav Atreya, Karl Pertsch, Tony Lee, Moo Jin Kim, Arhan Jain, Artur Kuramshin, Clemens Eppner, Cyrus Neary, Edward Hu, Fabio Ramos, et al. Roboarena: Distributed real-world evaluation of generalist robot policies. *arXiv preprint arXiv:2506.18123*, 2025.

Philip J. Ball, Jakob Bauer, Frank Belletti, Bethanie Brownfield, Ariel Ephrat, Shlomi Fruchter, Agrim Gupta, Kristian Holsheimer, Aleksander Holynski, Jiri Hron, Christos Kaplanis, Marjorie Limont, Matt McGill, Yanko Oliveira, Jack Parker-Holder, Frank Perbet, Guy Scully, Jeremy Shar, Stephen Spencer, Omer Tov, Ruben Villegas, Emma Wang, Jessica Yung, Cip Baetu, Jordi Berbel, David Bridson, Jake Bruce, Gavin Buttimore, Sarah Chakera, Bilva Chandra, Paul Collins, Alex Cullum, Bogdan Damoc, Vibha Dasagi, Maxime Gazeau, Charles Gbadamosi, Woohyun Han, Ed Hirst, Ashyana Kachra, Lucie Kerley, Kristian Kjems, Eva Knoepfel, Vika Koriakin, Jessica Lo, Cong Lu, Zeb Mehring, Alex Moufarek, Henna Nandwani, Valeria Oliveira, Fabio Pardo, Jane Park, Andrew Pierson, Ben Poole, Helen Ran, Tim Salimans, Manuel Sanchez, Igor Saprykin, Amy Shen, Sailesh Sidhwani, Duncan Smith, Joe Stanton, Hamish Tomlinson, Dimple Vijaykumar, Luyu Wang, Piers Wingfield, Nat Wong, Keyang Xu, Christopher Yew, Nick Young, Vadim Zubov, Douglas Eck, Dumitru Erhan, Koray Kavukcuoglu, Demis Hassabis, Zoubin Gharamani, Raia Hadsell, Aäron van den Oord, Inbar Mosseri, Adrian Bolton, Satinder Singh, and Tim Rocktäschel. Genie 3: A new frontier for world models. 2025.

Homanga Bharadhwaj, Debidatta Dwibedi, Abhinav Gupta, Shubham Tulsiani, Carl Doersch, Ted Xiao, Dhruv Shah, Fei Xia, Dorsa Sadigh, and Sean Kirmani. Gen2act: Human video generation in novel scenarios enables generalizable robot manipulation. *arXiv preprint arXiv:2409.16283*, 2024.

Kevin Black, Mitsuhiko Nakamoto, Pranav Atreya, Homer Walke, Chelsea Finn, Aviral Kumar, and Sergey Levine. Zero-shot robotic manipulation with pretrained image-editing diffusion models. *arXiv preprint arXiv:2310.10639*, 2023.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. $\pi 0$: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

Kevin Black, Manuel Y Galliker, and Sergey Levine. Real-time execution of action chunking flow policies. *arXiv preprint arXiv:2506.07339*, 2025.

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023a.

Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22563–22575, 2023b.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 37:24081–24125, 2024.

Can Cui, Pengxiang Ding, Wenxuan Song, Shuanghao Bai, Xinyang Tong, Zirui Ge, Runze Suo, Wanqi Zhou, Yang Liu, Bofang Jia, et al. Openhelix: A short survey, empirical analysis, and open-source dual-system vla model for robotic manipulation. *arXiv preprint arXiv:2505.03912*, 2025.

Sudeep Dasari, Frederik Ebert, Stephen Tian, Suraj Nair, Bernadette Bucher, Karl Schmeckpeper, Siddharth Singh, Sergey Levine, and Chelsea Finn. Robonet: Large-scale multi-robot learning. *arXiv preprint arXiv:1910.11215*, 2019.

Yilun Du, Sherry Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Josh Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. *Advances in Neural Information Processing Systems*, 36, 2024.

Yuqing Du, Ksenia Konyushkova, Misha Denil, Akhil Raju, Jessica Landon, Felix Hill, Nando De Freitas, and Serkan Cabi. Vision-language models as success detectors. *arXiv preprint arXiv:2303.07280*, 2023.

Frederik Ebert, Chelsea Finn, Sudeep Dasari, Annie Xie, Alex Lee, and Sergey Levine. Visual foresight: Model-based deep reinforcement learning for vision-based robotic control. *arXiv preprint arXiv:1812.00568*, 2018.

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

Chelsea Finn and Sergey Levine. Deep visual foresight for planning robot motion. In *2017 IEEE international conference on robotics and automation (ICRA)*, pp. 2786–2793. IEEE, 2017.

Chongkai Gao, Haozhuo Zhang, Zhixuan Xu, Zhehao Cai, and Lin Shao. Flip: Flow-centric generative planning as general-purpose manipulation world model. *arXiv preprint arXiv:2412.08261*, 2024.

Shenyuan Gao, Siyuan Zhou, Yilun Du, Jun Zhang, and Chuang Gan. Adaworld: Learning adaptable world models with latent actions. *arXiv preprint arXiv:2503.18938*, 2025.

Yanjiang Guo, Yucheng Hu, Jianke Zhang, Yen-Jen Wang, Xiaoyu Chen, Chaochao Lu, and Jianyu Chen. Prediction with action: Visual policy learning via joint denoising process. *Advances in Neural Information Processing Systems*, 37:112386–112410, 2024.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019.

Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. *arXiv preprint arXiv:2010.02193*, 2020.

Nicklas Hansen, Xiaolong Wang, and Hao Su. Temporal difference learning for model predictive control. *arXiv preprint arXiv:2203.04955*, 2022.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Alain Hore and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In *2010 20th international conference on pattern recognition*, pp. 2366–2369. IEEE, 2010.

Yucheng Hu, Yanjiang Guo, Pengchao Wang, Xiaoyu Chen, Yen-Jen Wang, Jianke Zhang, Koushil Sreenath, Chaochao Lu, and Jianyu Chen. Video prediction policy: A generalist robot policy with predictive visual representations. *arXiv preprint arXiv:2412.14803*, 2024.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. $pi\_\{0.5\}$: a vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025.

Joel Jang, Seonghyeon Ye, Zongyu Lin, Jiannan Xiang, Johan Bjorck, Yu Fang, Fengyuan Hu, Spencer Huang, Kaushil Kundalia, Yen-Chen Lin, et al. Dreamgen: Unlocking generalization in robot learning through neural trajectories. *arXiv e-prints*, pp. arXiv–2505, 2025.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.

Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

Shuang Li, Yihuai Gao, Dorsa Sadigh, and Shuran Song. Unified video action model. *arXiv preprint arXiv:2503.00200*, 2025a.

Yaxuan Li, Yichen Zhu, Junjie Wen, Chaomin Shen, and Yi Xu. Worldeval: World model as real-world robot policies evaluator. *arXiv preprint arXiv:2505.19017*, 2025b.

Junbang Liang, Ruoshi Liu, Ege Ozguroglu, Sruthi Sudhakar, Achal Dave, Pavel Tokmakov, Shuran Song, and Carl Vondrick. Dreamitate: Real-world visuomotor policy learning via video generation. *arXiv preprint arXiv:2406.16862*, 2024.

Yue Liao, Pengfei Zhou, Siyuan Huang, Donglin Yang, Shengcong Chen, Yuxin Jiang, Yue Hu, Jingbin Cai, Si Liu, Jianlan Luo, et al. Genie envisioner: A unified world foundation platform for robotic manipulation. *arXiv preprint arXiv:2508.05635*, 2025.

Jiaming Liu, Hao Chen, Pengju An, Zhuoyang Liu, Renrui Zhang, Chenyang Gu, Xiaoqi Li, Ziyu Guo, Sixiang Chen, Mengzhen Liu, et al. Hybridvla: Collaborative diffusion and autoregression in a unified vision-language-action model. *arXiv preprint arXiv:2503.10631*, 2025.

Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. Rdt-1b: a diffusion foundation model for bimanual manipulation. *arXiv preprint arXiv:2410.07864*, 2024.

Yecheng Jason Ma. Foundation reward models for general robot skill acquisition. In *Robotics: Science and Systems-Pioneers Workshop 2025*, 2025.

Anusha Nagabandi, Kurt Konolige, Sergey Levine, and Vikash Kumar. Deep dynamics models for learning dexterous manipulation. In *Conference on robot learning*, pp. 1101–1112. PMLR, 2020.

Junhyuk Oh, Xiaoxiao Guo, Honglak Lee, Richard L Lewis, and Satinder Singh. Action-conditional video prediction using deep networks in atari games. *Advances in neural information processing systems*, 28, 2015.

Karl Pertsch, Kyle Stachowicz, Brian Ichter, Danny Driess, Suraj Nair, Quan Vuong, Oier Mees, Chelsea Finn, and Sergey Levine. Fast: Efficient action tokenization for vision-language-action models. *arXiv preprint arXiv:2501.09747*, 2025.

Julian Quevedo, Percy Liang, and Sherry Yang. Evaluating robot policies in a world model. *arXiv preprint arXiv:2506.00613*, 2025.

Hengkai Tan, Yao Feng, Xinyi Mao, Shuhe Huang, Guodong Liu, Zhongkai Hao, Hang Su, and Jun Zhu. Anypos: Automated task-agnostic actions for bimanual manipulation. *arXiv preprint arXiv:2507.12768*, 2025.

1X World Model Team. 1x world model: Evaluating bits, not atoms. 2025. URL `https://www.1x.tech/1x-world-model.pdf`.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018.

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

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5294–5306, 2025.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Junjie Wen, Yichen Zhu, Jinming Li, Zhibin Tang, Chaomin Shen, and Feifei Feng. Dexvla: Vision-language model with plug-in diffusion expert for general robot control. *arXiv preprint arXiv:2502.05855*, 2025.

Jialong Wu, Shaofeng Yin, Ningya Feng, Xu He, Dong Li, Jianye Hao, and Mingsheng Long. ivideogpt: Interactive videogpts are scalable world models. *Advances in Neural Information Processing Systems*, 37:68082–68119, 2024.

Philipp Wu, Alejandro Escontrela, Danijar Hafner, Pieter Abbeel, and Ken Goldberg. Daydreamer: World models for physical robot learning. In *Conference on robot learning*, pp. 2226–2240. PMLR, 2023.

Annie Xie, Frederik Ebert, Sergey Levine, and Chelsea Finn. Improvisation through physical understanding: Using novel objects as tools with visual foresight. *arXiv preprint arXiv:1904.05538*, 2019.

Mengjiao Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. *arXiv preprint arXiv:2310.06114*, 1(2):6, 2023.

Jianke Zhang, Yanjiang Guo, Yucheng Hu, Xiaoyu Chen, Xiang Zhu, and Jianyu Chen. Up-vla: A unified understanding and prediction model for embodied agent. *arXiv preprint arXiv:2501.18867*, 2025.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 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.

Qingqing Zhao, Yao Lu, Moo Jin Kim, Zipeng Fu, Zhuoyang Zhang, Yecheng Wu, Zhaoshuo Li, Qianli Ma, Song Han, Chelsea Finn, et al. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 1702–1713, 2025.

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

Ruijie Zheng, Jing Wang, Scott Reed, Johan Bjorck, Yu Fang, Fengyuan Hu, Joel Jang, Kaushil Kundalia, Zongyu Lin, Loic Magne, et al. Flare: Robot learning with implicit world modeling. *arXiv preprint arXiv:2505.15659*, 2025.

Zhide Zhong, Haodong Yan, Junfeng Li, Xiangchen Liu, Xin Gong, Wenxuan Song, Jiayi Chen, and Haoang Li. Flowvla: Thinking in motion with a visual chain of thought. *arXiv preprint arXiv:2508.18269*, 2025.

Chuning Zhu, Raymond Yu, Siyuan Feng, Benjamin Burchfiel, Paarth Shah, and Abhishek Gupta. Unified world models: Coupling video and action diffusion for pretraining on large robotic datasets. *arXiv preprint arXiv:2504.02792*, 2025.

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

Code can be found in the Supplementary Materials.

More videos can be found at the anonymous website: `https://sites.google.com/view/ctrl-world`.

A More Details for World Model Learning

Model Architecture. Our world model closely follows the architecture of Stable Video Diffusion (SVD) (Blattmann et al., 2023a), and initializes from the SVD pretrained checkpoint. The only newly initialized component is a 3-layer MLP that projects 7-dimensional Cartesian-space actions into a 1024-dimensional latent embedding.

The input images are first encoded by a VAE with a spatial downsampling ratio of 8×8. In practice, we use $k = 7$ history frames, each perturbed with independent random noise to improve robustness. We set the action conditioning window to be one second, corresponding to 15 action steps. To reduce GPU memory consumption, we transform these 15 actions in the Cartesian space (see Section B) and temporally downsample them to 5 steps before feeding them into the model.

Each frame contains three 192×320 images, which are encoded into latent features of shape 24×40. The resulting total input token shape is $B \times (7 + 5) \times (3 \times 24 \times 40)$, which is then processed by the spatial-temporal transformer backbone.

Training Datasets. We use all 95k trajectories from the DROID dataset. For each training step, we randomly sample a trajectory and then uniformly sample a frame within that trajectory as the current frame. We then retrieve memory frames by sampling backward in time and set the model's prediction target to be the subsequent future frames.

Training Process. We train the model on 2×8 H100 GPUs with a total batch size of 64. The learning rate is set to be 1e-5, and we train for 100k steps, which takes approximately 2–3 days to complete.

B More Details for Policy Evaluation

Details on interaction between policy and world model. We directly use the official π_0-DROID, π_0-FAST-DROID, and $\pi_{0.5}$-DROID policies from `https://github.com/Physical-Intelligence/openpi` to interact with Ctrl-World. To the best of our knowledge, Ctrl-World is the first world model that enables policy-in-the-loop interactions between state-of-the-art VLA model. These open-sourced policies take joint angles and two views of camera as input and output joint velocities. In contrast, our world model conditions on the end-effector pose in Cartesian space. To bridge this mismatch, we train an *adapter* on the DROID dataset that maps the current joint angles q_t^{joint} and predicted joint velocities $a_{t+1:t+H}^{\text{jv}}$ into future joint configurations $q_{t+1:t+H}^{\text{joint}}$. We then apply Franka Panda forward kinematics (FK) to convert these joint configurations into Cartesian-space poses $q_{t+1:t+H}^{\text{cartesian}}$. The adapter is implemented as a simple two-layer MLP.

The overall process is as follows: given the current joint configuration q_t^{joint}, multi-view observation o_t, and language instruction l, the policy outputs H-step joint velocities:

$$a_{t+1:t+H}^{\text{jv}} = \pi(q_t^{\text{joint}}, o_t, l).$$

These are passed through the adapter to predict future joint configurations, followed by FK to compute Cartesian poses:

$$q_{t+1:t+H}^{\text{joint}} = \text{Adapter}(q_t^{\text{joint}}, a_{t+1:t+H}^{\text{jv}}), \qquad q_{t+1:t+H}^{\text{cartesian}} = FK(q_{t+1:t+H}^{\text{joint}}).$$

Finally, the world model predicts the next H frames conditioned on the current observation, the calculated Cartesian poses, and the history Cartesian poses:

$$o_{t+1:t+H} = WM(o_t, q_{t+1:t+H}^{\text{cartesian}}, q_{\text{history}}^{\text{cartesian}}).$$

This setup enables fully autoregressive rollouts, allowing the official π_0-DROID, π_0-FAST-DROID, and $\pi_{0.5}$-DROID policies and Ctrl-World to interact seamlessly in imagination space.

Task	Method	Instruction Following		Success Rate	
		Real world	World Model	Real world	World Model
Pick-Place	π_0	0.75	0.60	0.70	0.55
	π_0-fast	0.90	0.75	0.85	0.70
	$\pi_{0.5}$	0.90	0.80	0.85	0.70
Fold-Towel	π_0	0.40	0.20	0.40	0.20
	π_0-fast	0.55	0.45	0.55	0.40
	$\pi_{0.5}$	0.80	0.75	0.75	0.65
Drawer	π_0	0.05	0.00	0.05	0.00
	π_0-fast	0.20	0.20	0.15	0.05
	$\pi_{0.5}$	0.80	0.65	0.80	0.30
Wipe-table	π_0	0.20	0.10	0.20	0.00
	π_0-fast	0.50	0.35	0.50	0.35
	$\pi_{0.5}$	0.80	0.70	0.80	0.65
Close-laptop	π_0	0.15	0.20	0.15	0.00
	π_0-fast	0.60	0.50	0.50	0.00
	$\pi_{0.5}$	0.80	0.70	0.70	0.05
Pull-tissue	π_0	0.30	0.20	0.30	0.20
	π_0-fast	0.10	0.0	0.10	0.0
	$\pi_{0.5}$	0.85	0.60	0.75	0.50
Stack	π_0	0.75	0.45	0.60	0.40
	π_0-fast	0.80	0.70	0.70	0.55
	$\pi_{0.5}$	0.80	0.65	0.65	0.55

Table 3: Comparison of instruction-following and success rate across methods and tasks.

Breakdown for policy evaluation. We present the instruction-following and low-level execution success rates in Table 3.

Task details and criterion. In our experiments, we use human annotators to evaluate whether each trajectory is a success or a failure. Although this evaluation process can be automated in the future using large vision-language reward models, our focus in this paper is on the world model itself, so we rely on human preference as the reward signal. We provide clear criteria to determine whether a trajectory merely follows the instruction or achieves full task success:

- **Pick-place**: Several objects and receptacles are placed on the tabletop. The instruction is of the form "Pick up A and place in B." A trajectory is considered to follow the instruction if the policy attempts to grasp the correct object A. It is considered a success if object A is successfully placed into the target receptacle B.
- **Fold the Towel:** A towel is lying flat on the table, with other objects possibly present. The instruction is "Fold the towel." A trajectory is considered to follow the instruction if the gripper moves to the towel's edge and attempts to lift and fold it. A trajectory is considered successful if the towel's surface area becomes half in the end.
- **Drawer:** The instruction is to "Place object A into drawer". A trajectory follows the instruction if the robot attempts to place object A inside the drawer. It is a success if object A is eventually placed in the drawer.
- **Wipe Table:** The instruction is to wipe the table surface. A trajectory follows the instruction if the gripper makes contact with the towel and moves in a sweeping motion. It is considered successful if a large portion of the table is covered by the sweeping motion.
- **Close Laptop:** The instruction is to close an open laptop. A trajectory follows the instruction if the gripper approaches the laptop lid. It is considered successful if the lid is fully closed.
- **Pull Tissue:** The instruction is to pull a tissue from a tissue box. A trajectory follows the instruction if the gripper approaches the tissue slot and pinches a tissue. It is considered successful if at least one tissue is fully extracted.

- **Stack:** The instruction is to stack one object on top of another. A trajectory follows the instruction if the gripper lifts the correct object. It is a success if the object is placed stably on top of the target object.

C More Details for Policy Improvement

Finetuning Process. We finetune $\pi_{0.5}$-DROID policy based on official codebase `https://github.com/Physical-Intelligence/openpi`. We finetune the pretrained checkpoint on our synthetic dataset for 2k steps on 4 H100 GPUs.

Task Descriptions:

- **Spatial Understanding Tasks:** 2–6 random objects are placed on the table. The policy is instructed to pick an object at a specified spatial location and place it in the box. Example instructions include: "Pick the object on the top-right side and place it in the box" or "Place the object on the far-left side into the box."
- **Shape Understanding Tasks:** 2–3 random objects are placed on the table, where some share the same attributes but differ in size. The policy must distinguish objects based on the size. Example instruction: "Pick the larger red block and place it in the box."
- **Towel-Folding with Directions:** A towel and other distractor are placed on the table, and the policy is given instructions specifying a particular folding direction (e.g., "Fold the towel from left to right").
- **Novel Objects:** We introduce unseen objects such as a glove and a stapler which Pretrained policy can not identify very well.

Detailed success rate. We provide detailed task success rates inside each categories:

	Left	Right	Bottom	Top	Left Top	Left Bottom	Right Top	Right Bottom	Average
Base Policy	0.50	0.45	0.30	0.45	0.15	0.20	0.05	0.20	0.2875
After Post-Training	0.85	0.90	1.00	0.80	0.85	0.90	0.90	0.80	0.875

Table 4: Policy improvement (Spatial Understanding).

	Big Left	Big Right	Small Left	Small Right	Average
Base Policy	0.40	0.45	0.40	0.50	0.4374
After Post-Training	0.85	0.95	0.95	0.90	0.9125

Table 5: Policy improvement (Shape understanding).

	Towel-1	Towel-2	Towel-3	Towel-4	Average
Base Policy	0.60	0.50	0.55	0.65	0.575
After Post-Training	0.75	0.8	0.85	0.80	0.80

Table 6: Policy improvement (Towel folding with direction).

	Novel-obj-glove	Novel-obj-stapler	Average
Base Policy	0.20	0.30	0.25
After Post-Training	0.80	0.70	0.75

Table 7: Policy improvement (Novel object).