CO-EVOLVING LATENT ACTION WORLD MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Adapting pre-trained video generation models into controllable world models via *latent actions* is a promising step towards creating generalist world models. The dominant paradigm adopts a two-stage approach that trains latent action model (LAM) and the world model separately, resulting in redundant training and limiting their potential for co-adaptation. A conceptually simple and appealing idea is to directly replace the forward dynamic model in LAM with a powerful world model and training them jointly, but it is non-trivial and prone to representational collapse. In this work, we propose **CoLA-World**, which for the first time successfully realizes this synergistic paradigm, resolving the core challenge in joint learning through a critical warm-up phase that effectively aligns the representations of the from-scratch LAM with the pre-trained world model. This unlocks a co-evolution cycle: the world model acts as a knowledgeable tutor, providing gradients to shape a high-quality LAM, while the LAM offers a more precise and adaptable control interface to the world model. Empirically, CoLA-World matches or outperforms prior two-stage methods in both video simulation quality and downstream visual planning, establishing a robust and efficient new paradigm for the field.

1 INTRODUCTION

A prevailing goal in artificial intelligence is the creation of a generalist agent capable of acting across a multitude of environments and embodiments. Central to this vision is the concept of a *world model* (Sutton, 1990; Ha & Schmidhuber, 2018), an internal simulator of the environment that allows an agent to plan and learn through imagination. An ideal world model would be universal, leveraging vast priors about world physics and dynamics, and adaptable with minimal data to any specific downstream task. While large-scale video generative models (OpenAI, 2024; Blattmann et al., 2023) have emerged as powerful candidates for such general-purpose simulators due to their rich pre-trained knowledge, a fundamental challenge remains: how to interactively control the generation. The heterogeneity of action spaces across different domains, from the continuous torques of a robot arm to the discrete button presses of a game console, prohibits the direct use of real actions for finetuning a video generative model to a single, universal world model.

To bridge this gap, Latent Action Models (LAMs) have shown great promise (Schmidt & Jiang, 2023; Bruce et al., 2024; Ye et al., 2025). By inferring abstract actions directly from visual observations, LAMs provide a unified, embodiment-agnostic interface for controlling a world model. This paradigm opens an exciting direction: pre-training a single, general-purpose world model conditioned on a universal latent action space (Bruce et al., 2024; NVIDIA et al., 2025; Gao et al., 2025). To integrate LAMs with world models, existing works typically adopt a two-stage approach: first training a LAM on action-free videos, usually with a small inverse dynamics model (IDM) and a forward dynamics model (FDM) trained from scratch, and then freezing the IDM to supply latent actions for training a larger world model.

However, this two-stage approach faces several issues. First, the FDM and the world model are essentially both performing next-observation prediction, rendering the overall framework redundant. Second, the pipeline forces the world model to rely on a fixed, static latent action space, preventing the latent actions from adapting as world model training progresses. One question naturally arises:

Can we replace the FDM with the world model?

At first glance, this might seem like a straightforward modification, but our experiments show that naively training the IDM and world model together can easily lead to collapse.

In this work, we explore this question and provide an affirmative answer. We propose **CoLA-World**, a training pipeline that enables the synergistic co-evolution of latent action learning and world modeling. We first observe that, whether the IDM is initialized from scratch or from a pre-trained one, direct joint training with the world model leads to collapse. This suggests that the IDM is not well aligned with the pre-trained weights of the world model.

To address this, before switching to joint training, CoLA-World introduces a warm-up phase in which the world model is kept frozen and only supplies gradients to update the IDM. This greatly stabilizes subsequent joint training and enables the IDM and world model to co-evolve effectively. On one hand, the powerful world model carries prior knowledge of plausible physics and visual dynamics inherited from a pre-trained video generation model. It acts as an active tutor, providing gradients that guide the from-scratch IDM toward higher-quality latent actions. On the other hand, as the IDM learns to produce a more informative latent action space, it in turn offers the world model a clearer and more precise control interface.

We evaluate our method on a large-scale dataset consisting of human egocentric and robot manipulation videos. Compared to baseline two-stage methods, CoLA-World learns higher-quality latent actions and achieves stronger world model prediction performance. We further provide empirical evidence that co-evolution in the joint-training phase is crucial, as it enables both latent action learning and world modeling to outperform setups where either component is fixed. Finally, we assess the adaptability of the learned latent-action-based world models to out-of-distribution real-action control interfaces, showing that the joint training enabled by our method is key to improving both video prediction quality and downstream visual planning.

In summary, our main contributions are:

- We propose CoLA-World, the first framework that successfully enables joint training of a latent action model with a pre-trained video-generation-based world model.
- Compared to prior two-stage methods, CoLA-World's joint latent action learning and world modeling yield a higher-quality latent action space and a world model with stronger controllability and sample efficiency, improving both video simulation and downstream visual planning.
- We show that CoLA-World's joint training exhibits synergistic co-evolution: the improving world model and LAM mutually reinforce each other, creating a tightly coupled system that drives superior adaptability.

2 Related Work

Latent Action Learning Latent actions have recently emerged as a promising approach for behavior pre-training on action-free data. Early methods such as FICC (Ye et al., 2023) and LAPO (Schmidt & Jiang, 2023) adopt the IDM–FDM framework, where latent actions are discovered through a next-frame reconstruction objective. Genie (Bruce et al., 2024) scales this framework to large transformer-based architectures, focusing on latent-action-driven world model prediction in addition to policy learning. A few works (Ye et al., 2025; NVIDIA et al., 2025; Bu et al., 2025; Chen et al., 2025) have also explored the utility of latent action learning in embodied agents, particularly in the vision–language–action setting. Our work differs from prior approaches in that we leverage a pre-trained video generation model to co-evolve latent action learning and world modeling, a direction that has not been explored before.

Latent-action-based World Models While the FDM in the latent action model can be interpreted as a world model, most works do not explicitly focus on future prediction abilities, with the exception of (Cui & Gao, 2023). However, the prediction quality of FDMs is generally lower than that of high-capacity video-generation-based world models. Recently, Genie (Bruce et al., 2024) trained a separate decoder-only MaskGIT (Chang et al., 2022) as the world model, conditioned on a fixed latent action space learned beforehand. AdaWorld (Gao et al., 2025) is the work most closely related to ours, adopting a similar two-stage approach as Genie but using a diffusion-based video model and extending discrete latent actions to continuous ones. Other efforts, such as AD3 (Wang et al., 2024b) and PreLAR (Zhang et al., 2024), integrate latent action learning with dynamics and policy training in a Dreamer-style (Hafner et al., 2021) architecture trained from scratch, rather than leveraging the benefits of large-scale pre-trained video generation models.

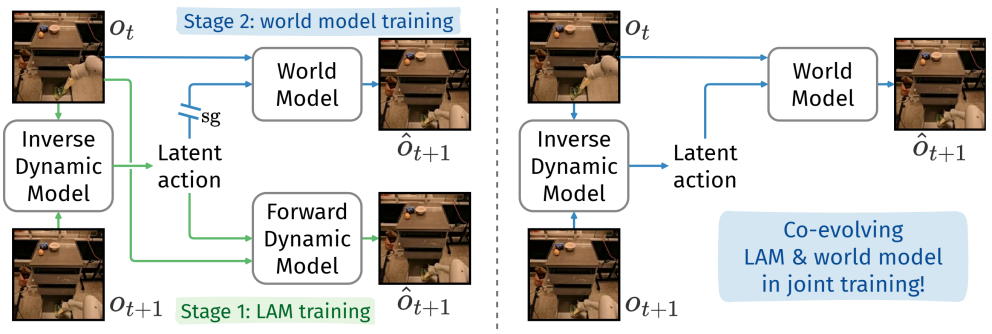

Figure 1: **(a)** Prior works use a two-stage pipeline: learn a latent action model (LAM), then fix it to train the world model. **(b)** We propose a one-stage pipeline, directly using the world model as the forward dynamics model and backpropagating gradients through latent actions.

Finetuning Pre-trained Video Generation Model as World Models Our work is also related to efforts that fine-tune pre-trained video generation models into controllable world models by adding action conditioning. Except for AdaWorld (Gao et al., 2025) discussed above, most works in this line assume a pre-specified action space. AVID (Rigter et al., 2025) introduces a lightweight adapter on top of a frozen video generation model for action conditioning and world modeling. IRASim (Zhu et al., 2024) uses adaptive layer normalization (Peebles & Xie, 2023) to incorporate actions, analogous to how text prompting is conditioned. Following IRASim, DWS (He et al., 2025) proposes a more granular action conditioning mechanism along with other improvements for world modeling. Vid2World (Huang et al., 2025a) focuses on challenges of temporal causality in adapting video diffusion models to world models, while EnerVerse-AC (Jiang et al., 2025) adds action conditioning to a robotics foundation model (Huang et al., 2025b) for manipulation tasks.

3 Method

3.1 World Models with Latent Actions

We focus on training a world model to predict the next observation o_{t+1} based on the current observation o_t and a *latent action* z_t, modeling the distribution $p(o_{t+1} \mid o_t, z_t)$. Unlike pre-specified actions, such as keyboard or mouse inputs in video games, latent actions are learned entirely from observational data. This allows us to pre-train world models on large-scale, action-free video data.

As mentioned in the introduction, previous works (Bruce et al., 2024; Gao et al., 2025) typically adopt a two-stage process, training a latent action model (LAM) prior to world model training. The LAM consists of an inverse dynamics model (IDM) and a forward dynamics model (FDM). Specifically, the IDM f_{inv} takes the current observation o_t and the next observation o_{t+1} as input and outputs a latent action z_t, while the FDM f_{fwd} takes o_t and z_t to predict the next observation $\hat{o}_{t+1}$. LAM is trained by minimizing the reconstruction loss between $\hat{o}_{t+1}$ and o_{t+1}, *i.e.*,

$$\mathcal{L}_{\text{LAM}} = \|o_{t+1} - f_{\text{fwd}}(o_t, f_{\text{inv}}(o_t, o_{t+1}))\|. \tag{1}$$

To prevent trivial solutions, a bottleneck is often applied to the latent action space, forcing the latent actions to compactly encode the most meaningful changes between o_t and o_{t+1}. Once trained, the IDM is frozen and used to extract latent action labels for observation sequences. Previous works then train a separate world model to capture $p(o_{t+1} \mid o_t, z_t)$, typically employing a much higher-capacity model than the LAM. The complete pipeline is illustrated in Figure 1(a).

However, one may immediately notice that the FDM and the world model perform exactly the same task: predicting o_{t+1} based on o_t and z_t. Our idea is to replace the FDM with the world model, reducing the two-stage training into a single joint training framework that performs dynamics learning and latent action learning simultaneously in an end-to-end fashion, as illustrated in Figure 1(b). Such a framework not only enables a more elegant model design and efficient training but also allows the co-evolution of latent actions and the world model. The powerful world model can provide gradients that help the IDM learn higher-quality latent actions, while the IDM produces a more informative latent action space, offering the world model a clearer control interface.

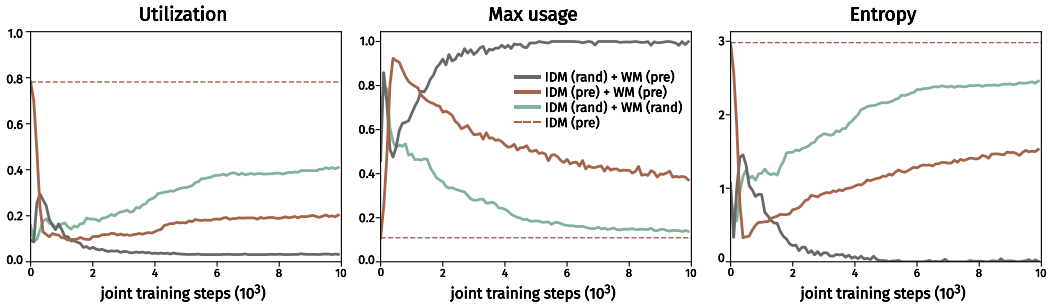

Figure 2: Latent action codebook metrics during joint training of the IDM and world model. "rand" indicates random initialization, while "pre" indicates initialization from pre-trained weights. The dashed line shows the codebook metrics of the pre-trained IDM. All three subplots share the same legend, shown only in the middle panel for clarity.

While this idea may seem simple, we show in the next subsection that naively training the inverse dynamics model and the world model together can easily collapse. One might also argue that the FDM is essentially a world model and could be used to roll out future predictions. Empirically, however, we find that the FDM produces much lower-quality predictions than a separately trained world model. We believe this explains why previous works adopt a two-stage approach. To the best of our knowledge, no prior work has successfully attempted this type of joint training.

3.2 Taming the Fragility of Joint Training

Following prior work (Bruce et al., 2024; Gao et al., 2025), we instantiate the IDM in Figure 1(b) as an ST-Transformer (Xu et al., 2020), followed by vector quantization (Van Den Oord et al., 2017) to produce discrete latent actions. For the world model, we adopt OpenSora (Zheng et al., 2024), a high-performing open-source diffusion-based video generative model. We choose OpenSora for its demonstrated effectiveness in the DWS method (He et al., 2025), where it was adapted for world modeling with pre-specified actions. Additional implementation details are deferred to Section 3.3.

When training the model, however, we observe that learning quickly collapses. As shown by the gray curve in Figure 2, the utilization rate of the VQ codebook for the latent actions drops to zero after an initial brief increase. At the same time, the maximum code usage rapidly rises to nearly 100%, indicating that the model collapses to using only a very small subset of latent actions. The concurrent drop of code entropy to zero further suggests that all codes in the codebook degenerate into a single dominant code. In contrast, a healthy latent action codebook should exhibit relatively high utilization and entropy, along with low maximum usage, as indicated by the dashed horizontal lines in Figure 2.

As we have seen, directly training a freshly initialized IDM jointly with a pre-trained world model leads to collapse. We hypothesize that this occurs because the powerful, pre-trained world model quickly learns to disregard the random and uninformative action signals provided by the from-scratch LAM. By relying on its own strong internal priors to minimize the prediction loss, the world model provides no structured, supervisory gradient back to the LAM, causing its representation to degenerate into a few dominant, uninformative codes. To further investigate the fragility of joint training, we next initialize the IDM using parameters from a reasonably well-trained latent action model (corresponding to the dashed horizontal lines in Figure 2). However, as the brown curve in Figure 2 shows, even though it starts from a favorable state, the codebook quickly deteriorates, leading to low utilization and entropy. Although it gradually improves later, the progress remains too slow to be practical.

Given that neither random nor guided initialization works, we hypothesize that the IDM is not well aligned with the pre-trained weights of the world model. To test this, we randomly initialized both the IDM and the world model and trained them jointly. As shown by the green curve in Figure 2, this setup does not collapse, supporting our hypothesis. To mitigate the instability while still taking advantage of powerful pre-trained video generation models, we propose a warm-up strategy: first train the IDM while keeping the world model frozen, then switch to joint training.

With this warm-up, the IDM is able to catch up with the world model, enabling stable joint training without collapse. As the dark blue curve in Figure 3 shows, the codebook metrics remain healthy

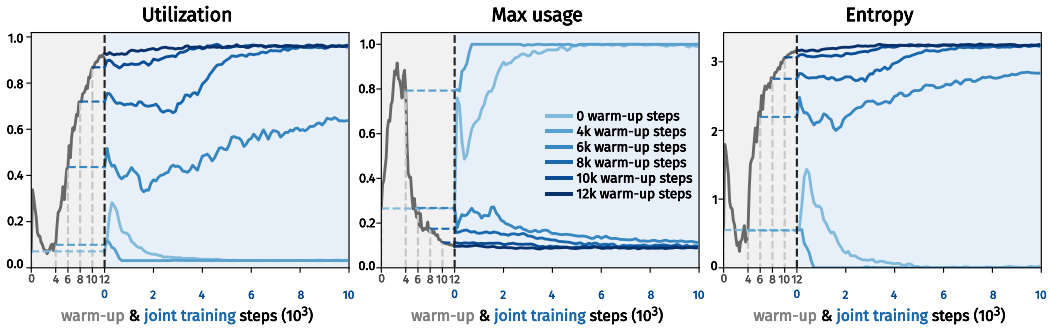

Figure 3: Latent action codebook metrics during warm-up and joint training. Different blue curves correspond to IDM initializations from warm-up checkpoints at various steps. All three subplots share the same legend, shown only in the middle panel for clarity.

under this scheme. We further varied the number of warm-up steps. Figure 3 shows that longer warm-up generally leads to more stable subsequent joint training, confirming that the IDM indeed undergoes a catch-up phase during warm-up. In practice, we choose a warm-up length that ensures stability while reserving as many steps as possible for end-to-end co-evolution.

After warm-up, we jointly train the IDM and world model end-to-end, allowing them to co-evolve and adapt to each other. The world model provides gradients that guide the IDM to learn higher-quality latent actions, while the IDM in turn produces a more informative latent action space for the world model. In Section 4, we present extensive experiments showing that this joint training strategy enhances both the quality of the learned latent actions and the performance of the world model.

3.3 Implementation Details

We elaborate on the key implementation details central to our joint training paradigm, focusing on the latent action conditioning mechanism and the end-to-end training process. Further information regarding model architectures and training details are deferred to the Appendix B.

Latent Action Conditioning. We integrate latent actions extracted by the IDM into the pre-trained OpenSora model via Adaptive Layer Normalization (AdaLN) (Peebles & Xie, 2023). The sequence of the latent actions is first processed by a from-scratch self-attention network to produce contextualized embeddings. These embeddings are then projected into action-specific scale, shift and gate parameters by a MLP, which are then fused via addition with the original modulation parameters derived from the diffusion timesteps, and applied at each LayerNorm layer within all the OpenSora blocks. This mechanism provides control signals to condition the denoising process on the latent actions.

Training Objective and Gradient Flow. The system is jointly optimized using a flow matching loss objective (Liu et al., 2022a) provided by the OpenSora model, which learns to predict the velocity needed to denoise the video latent. The warm-up and end-to-end training phases carefully manage the gradient flow generated by the loss. During warm-up, the pre-trained OpenSora model is frozen, and the loss is backpropagated through the action AdaLN parameters and solely update the action conditioning modules and the LAM components (IDM and VQ quantizer). Subsequently, in the end-to-end phase, we unfreeze the OpenSora world model and the unified gradient updates all components simultaneously. Crucially, this end-to-end gradient flow is the core mechanism for synergistic co-evolution.

4 Experiments

In this section, we conduct experiments to answer the following questions:

1. How does our joint training paradigm compare against the traditional two-stage approach in terms of LAM representation quality and world model video prediction performance?
2. What is the underlying mechanism of our paradigm's success? Do the LAM and the World Model truly engage in a synergistic co-evolution during joint learning?

3. Can the inherent advantages of our joint training paradigm translate into performance gains in practical real-action-based video simulation?
4. What is the ultimate efficacy of CoLA-World as a learned simulator for solving control tasks via visual planning?

4.1 Experimental Setup

Dataset We focus on learning latent-action-based world models for robotic manipulation that can adapt to diverse downstream embodiments and action spaces. Our training data consists of a large-scale mixture of human egocentric videos and robot manipulation videos. Importantly, the training process is entirely action-free: both the world model and the latent action model are learned purely from video. Full dataset details are provided in Appendix A.

Baselines We compare two training paradigms. **2-stage**: Following prior work, we first train a LAM (comprising an IDM, an FDM, and a VQ quantizer) from scratch. Then the LAM is frozen and its IDM and quantizer are used to provide latent actions for fine-tuning the world model, while the FDM is discarded. **Joint** (CoLA-World): Our joint learning paradigm begins with a brief warm-up phase to align the from-scratch LAM (IDM and quantizer) with the pre-trained world model, followed by full end-to-end (E2E) joint training. The architectures of the LAM and world model are identical across both paradigms. In the 2-stage setting, we train the LAM for 30K steps to ensure a high-quality representation. For joint training, we use an 8K warm-up phase (Figure 3), which provides a stable initialization while preserving budget for the E2E phase. Additional training details are provided in Appendix B. For clarity, we denote checkpoints by training budgets of their respective phases, *e.g.*, LAM30K + WM30K in 2-stage learning; Warm8K + E2E52K in joint learning.

Evaluation metrics. To assess the quality of the learned latent action, we employ a linear probing task, where a simple one-layer linear projection head is trained to predict the original real action from the frozen latent actions. Here we evaluate on L1 prediction loss to prevent potential outliers dominating the loss results. For the world model, we measure action-conditioned video generation quality using a suite of standard metrics: PSNR, SSIM, LPIPS and FVD. In the tables, LPIPS and SSIM scores are scaled ×100 for compact display.

4.2 Performance of the Jointly Learned LAM and World Model

Table 1: Linear probing loss across several robotics datasets (lower is better).

	Method	Bridge	RT-1	Kuka	Droid	AgiBot	LIBERO
2-stage	LAM30K	0.0827	**0.1191**	0.0741	0.1912	0.1035	**0.1614**
Joint	Warm8K + E2E22K	**0.0815**	0.1206	**0.0736**	**0.1911**	**0.0908**	0.1623

Latent Action Quality. We first evaluate the quality of the learned latent action representations via linear probing on six robotics datasets, including five from the Open X-Embodiment suite (Collaboration et al., 2023) and one out-of-distribution LIBERO dataset (Liu et al., 2023) unseen during training. As shown in Table 1, our CoLA-World yields a competitive latent action space, achieving lower probing loss on most datasets.

While the difference in probing loss appears marginal, this isolated metric does not fully capture the latent action representation's utility. The ultimate measure of a latent action's quality lies in its ability to effectively control the world model. As we will show, the world model guided by the jointly learned LAM significantly outperforms the two-stage baseline on LIBERO. This suggests that our co-evolved latent action space, while less amenable to linear probing, provides a more robust and effective control interface for world modeling.

World Model Simulation Performance. We then evaluate the latent-action-conditioned video prediction performance of the world model. Table 2 reports results across several in-distribution datasets (OXE, EgoCentric, AgiBot) and one out-of-distribution (LIBERO) dataset, comparing different training checkpoints. With the same total training budget of 60K steps, our joint training paradigm (Warm8K + E2E52K) consistently matches or surpasses the best two-stage method (LAM30K + WM30K) across all datasets. Notably, improvements are most pronounced on the

Table 2: Video prediction performance of the learned world models on different datasets.

Dataset	Method		PSNR ↑	SSIM ↑	LPIPS ↓	FVD ↓
OXE	2-stage	LAM30K + WM30K	22.34	81.16	13.17	291.30
		LAM8K + WM52K	21.91	80.76	13.79	296.64
	Joint	Warm8K + E2E52K	**22.57**	**81.40**	**12.79**	**278.90**
		Warm8K + E2E30K	22.26	81.06	13.26	289.37
EgoCentric	2-stage	LAM30K + WM30K	**23.80**	**83.68**	**12.90**	260.14
		LAM8K + WM52K	23.48	83.28	13.46	267.94
	Joint	Warm8K + E2E52K	23.69	83.52	13.08	**252.45**
		Warm8K + E2E30K	23.66	83.41	13.26	263.57
AgiBot	2-stage	LAM30K + WM30K	23.61	85.36	10.11	185.63
		LAM8K + WM52K	23.30	85.11	10.30	196.18
	Joint	Warm8K + E2E52K	**23.93**	**85.61**	**9.86**	**174.93**
		Warm8K + E2E30K	23.64	85.27	10.22	189.03
LIBERO	2-stage	LAM30K + WM30K	23.13	86.90	10.22	167.77
		LAM8K + WM52K	22.72	86.43	10.78	190.09
	Joint	Warm8K + E2E52K	**23.33**	**87.21**	**9.89**	**158.36**
		Warm8K + E2E30K	23.25	87.05	10.08	164.86

perceptually aligned FVD metric, indicating that our generated videos are not only pixel-accurate but also more temporally coherent and realistic.

Crucially, our paradigm also demonstrates superior sample efficiency. Our Warm8K + E2E30K model, with a substantially smaller budget, already approaches the performance of the fully trained LAM30K + WM30K 2-stage model and surpasses it on the out-of-distribution LIBERO dataset. This efficiency arises from the synergistic training, which avoids the redundant learning and static bottlenecks inherent in the 2-stage approach. Moreover, when the 2-stage method is given a similar total budget (LAM8K + WM52K *vs.* Warm8K + E2E52K), it is significantly outperformed, even lagging behind our less-trained Warm8K + E2E30K checkpoint due to its under-trained, static LAM. These results highlight that our joint training unlocks a higher performance ceiling with significantly fewer training steps. We provide latent action transfer results in Appendix D.2.

4.3 Evidence for Synergistic Co-evolution

Having shown the performance of our CoLA-World, we now turn to understanding the mechanism behind its success. To this end, we design two controlled ablation studies to dissect the bidirectional information flow and verify the presence of a virtuous cycle of mutual promotion.

An Evolving World Model as a Better Tutor for the LAM. To isolate the influence of the world model's own learning process on the LAM, we compare our Warmup + E2E method with a Pure warmup variant, where the LAM is trained using gradients from a frozen world model. We evaluate the resulting LAMs via linear probing loss on the LIBERO dataset, as shown in Figure 4(a). While the LAM guided by the static world model (Pure warmup) improves steadily, the LAM in our CoLA-World exhibits much faster reduction in probing loss once E2E training starts. This demonstrates that the supervisory signal from the world model evolves over time: as the world model refines its own understanding of the world's dynamics, the gradients it provides to the LAM become progressively more informative and causally sound. These results confirm that a concurrently improving world model acts as a effective tutor, enabling a better and more efficiently learned LAM.

An Evolving LAM as a Better Control Interface for the World Model. We then investigate the impact of a dynamically evolving LAM on the world model's video prediction performance. We compare our Warmup + E2E model against a variant where the LAM is frozen after the same initial warmup phase and only the world model is fine-tuned subsequently. As shown in Figure 4(b), the world model paired with a frozen LAM improves initially but quickly plateaus. By contrast, when paired with a continuously improving LAM during E2E training, the world model achieves substantially higher video generation quality. This demonstrates that a static latent action space

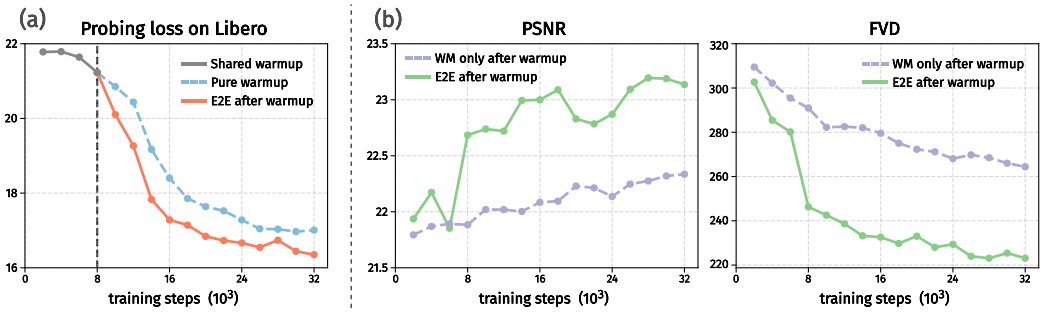

Figure 4: Evidence of synergistic co-evolution. The LAM's probing loss drops faster when the world model is co-evolving **(a)**, while the world model achieves higher video prediction performance as the LAM improves **(b)**.

imposes a performance bottleneck, whereas a dynamically evolving LAM provides a progressively more precise control interface, unlocking the world model's full predictive potential.

The Virtuous Cycle of Co-evolution. These two experiments provide evidence for a virtuous cycle of synergistic co-evolution: an improving world model better shapes the latent action representation, which in turn enables more effective world modeling. This dynamic co-evolution creates a deeply coupled and intrinsically consistent system. As shown in the following section, this property underlies our model's superior performance on downstream adaptation tasks.

4.4 Adaptation for Real-Action-Based Simulation

A key promise of latent-action-based world models is their adaptability to diverse, real-action control interfaces. We evaluate this capability by adapting our world model to new, out-of-distribution robotic environments including LIBERO and RoboDesk (Kannan et al., 2021).

Adaptation and Evaluation Protocol For each downstream dataset, we follow Gao et al. (2025) and first train a lightweight two-layer MLP adapter to map the dataset's real actions to the latent actions. Subsequently, we fine-tune the world models for 3K steps. Crucially, this fine-tuning is performed using ground-truth latent actions (GT-LAM), which are extracted from the downstream videos by the frozen learned LAM. This ensures the world model learns the new environment's dynamics from the clean supervisory signal, consistent with its pre-training. Finally, we evaluate the fine-tuned world model in two distinct modes: (a) using the same GT-LAM to assess the ideal performance ceiling after domain-specific finetuning, and (b) using the trained adapter to translate real actions into latent actions and assess the world model's practical, real-action-based video prediction performance.

Results and Analysis. To evaluate our paradigm's efficiency, we compare our jointly trained Warm8K + E2E30K checkpoint against the more extensively trained LAM30K + WM30K two-stage model. Despite using a smaller training budget, Table 3 shows that CoLA-World clearly outperforms the two-stage baseline. In GT-LAM evaluation, it already demonstrates an advantage, indicating that the jointly trained world model provides a stronger foundation for learning dynamics in unseen environments.

Moreover, the performance gap between CoLA-World and the two-stage baseline becomes more pronounced when evaluated with real actions, particularly on the FVD metric. This reflects a fundamental distinction in how the LAM and world model interact under the two paradigms. The two-stage model, fine-tuned on a fixed GT-LAM distribution, becomes rigidly calibrated to this static representation. When faced with biased latent actions from an imperfect adapter, the world model struggles to interpret these out-of-distribution signals, leading to a substantial performance drop.

By contrast, our world model co-evolves with a dynamically improving LAM, continually adapting to a smoothly changing latent action landscape. This process endows the world model with a more smooth and robust utilization of the latent action space, making it more resilient to the adapter's imperfections, correctly interpreting its biased outputs as functionally equivalent to the ground-truth signals. This intrinsic consistency allows CoLA-World to generalize effectively from ideal training

Table 3: Video prediction performance of the finetuned world models, taking latent actions inferred by the LAM or translated from the real actions by the learned adapters as conditions.

Dataset	Action Type	Method	PSNR ↑	SSIM ↑	LPIPS ↓	FVD ↓
LIBERO	GT-LAM	LAM30K + WM30K	25.51	89.55	7.41	**73.54**
		Warm8K + E2E30K	**25.85**	**89.82**	**7.31**	74.65
	Real Action	LAM30K + WM30K	22.45	86.96	9.56	115.45
		Warm8K + E2E30K	**22.68**	**87.15**	**9.27**	**93.68**
RoboDesk	GT-LAM	LAM30K + WM30K	24.21	86.99	**7.41**	120.51
		Warm8K + E2E30K	**24.29**	**87.04**	7.57	**120.26**
	Real Action	LAM30K + WM30K	20.03	83.33	10.64	188.82
		Warm8K + E2E30K	**21.37**	**84.67**	**8.90**	**169.70**

Table 4: Visual planning success rate on RoboDesk in the VP^2 benchmark.

Method	Upright Block	Push Slide	Flat Block	Push Drawer	Average
2-stage	20.0%	4.44%	1.11%	2.22%	6.94%
Joint	37.78%	6.11%	3.33%	5.25%	13.12%

signals to practical real-world control interfaces. Furthermore, as shown by a quantitative analysis of codebook metrics in Appendix D.1, the latent action space learned through joint training proves robust to the potential representation collapse observed in the two-stage approach during downstream adaptation, preserving its diversity and thus validating its strong generalization performance.

4.5 Visual Planning

To evaluate the final utility of our world model for downstream control, we assess the planning performance of our adapted world models using the VP^2 benchmark (Tian et al., 2023). We take the CoLA-World and two-stage models previously fine-tuned on the RoboDesk dataset and evaluate their ability to solve four challenging manipulation tasks using a sampling-based Model Predictive Control planner. The results, summarized in Table 4, indicate that our CoLA-World paradigm demonstrates a clear advantage over the two-stage approach, especially on Upright Block task. This confirms that the superior simulation quality demonstrated in Section 4.4 translates into more reliable prediction results for the planner, leading to more effective control.

On several complex tasks, both methods exhibited low performance, underscoring the inherent difficulty of these high-precision manipulation problems for any planner relying purely on a learned visual model. Nevertheless, the consistent and sometimes substantial performance gains achieved by CoLA-World on the tractable tasks strongly validate our joint training methodology as a more effective foundation for real-world control applications.

5 Conclusion, Limitation and Future Work

In this work, we introduce CoLA-World, the first framework to successfully realize the synergistic joint training of a latent action model with a pre-trained video-generation-based world model. A critical warmup phase resolves the inherent instability of this approach, enabling co-evolution between latent action learning and world modeling. Our experiments show that CoLA-World significantly outperforms previous two-stage methods in both simulation quality and downstream planning. A potential limitation is that the world model's performance depends on the pre-trained video generation model and requires substantial computational resources; however, this can be mitigated with more efficient models, and our paradigm is broadly applicable for injecting latent action conditioning. Future directions include evaluating the learned latent actions in vision-language-latent-action settings (Chen et al., 2025; Bu et al., 2025) for robotic manipulation policy training, and scaling our framework to train foundational world models on larger video datasets for broader adaptability.

REPRODUCIBILITY STATEMENT

We are committed to ensuring the reproducibility of our results. All datasets used in our experiments are publicly available, with detailed descriptions provided in Appendix A. Comprehensive information on model architectures and training protocols can be found in Appendix B. Our code is available in an anonymous repository for review at `https://anonymous.4open.science/r/CoLA-World`, and model checkpoints will be released upon publication.

REFERENCES

AgiBot-World-Contributors, Qingwen Bu, Jisong Cai, Li Chen, Xiuqi Cui, Yan Ding, Siyuan Feng, Shenyuan Gao, Xindong He, Xu Huang, Shu Jiang, Yuxin Jiang, Cheng Jing, Hongyang Li, Jialu Li, Chiming Liu, Yi Liu, Yuxiang Lu, Jianlan Luo, Ping Luo, Yao Mu, Yuehan Niu, Yixuan Pan, Jiangmiao Pang, Yu Qiao, Guanghui Ren, Cheng Ruan, Jiaqi Shan, Yongjian Shen, Chengshi Shi, Mingkang Shi, Modi Shi, Chonghao Sima, Jianheng Song, Huijie Wang, Wenhao Wang, Dafeng Wei, Chengen Xie, Guo Xu, Junchi Yan, Cunbiao Yang, Lei Yang, Shukai Yang, Maoqing Yao, Jia Zeng, Chi Zhang, Qinglin Zhang, Bin Zhao, Chengyue Zhao, Jiaqi Zhao, and Jianchao Zhu. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. *arXiv preprint arXiv: 2503.06669*, 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*, 2023.

Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *International Conference on Machine Learning*, pp. 4603–4623. PMLR, 2024.

Qingwen Bu, Yanting Yang, Jisong Cai, Shenyuan Gao, Guanghui Ren, Maoqing Yao, Ping Luo, and Hongyang Li. Univla: Learning to act anywhere with task-centric latent actions. *arXiv preprint arXiv:2505.06111*, 2025.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11315–11325, June 2022.

Xiaoyu Chen, Hangxing Wei, Pushi Zhang, Chuheng Zhang, Kaixin Wang, Yanjiang Guo, Rushuai Yang, Yucen Wang, Xinquan Xiao, Li Zhao, Jianyu Chen, and Jiang Bian. villa-x: Enhancing latent action modeling in vision-language-action models. *arXiv preprint arXiv: 2507.23682*, 2025.

Open X-Embodiment Collaboration, Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, Albert Tung, Alex Bewley, Alex Herzog, Alex Irpan, Alexander Khazatsky, Anant Rai, Anchit Gupta, Andrew Wang, Andrey Kolobov, Anikait Singh, Animesh Garg, Aniruddha Kembhavi, Annie Xie, Anthony Brohan, Antonin Raffin, Archit Sharma, Arefeh Yavary, Arhan Jain, Ashwin Balakrishna, Ayzaan Wahid, Ben Burgess-Limerick, Beomjoon Kim, Bernhard Schölkopf, Blake Wulfe, Brian Ichter, Cewu Lu, Charles Xu, Charlotte Le, Chelsea Finn, Chen Wang, Chenfeng Xu, Cheng Chi, Chenguang Huang, Christine Chan, Christopher Agia, Chuer Pan, Chuyuan Fu, Coline Devin, Danfei Xu, Daniel Morton, Danny Driess, Daphne Chen, Deepak Pathak, Dhruv Shah, Dieter Büchler, Dinesh Jayaraman, Dmitry Kalashnikov, Dorsa Sadigh, Edward Johns, Ethan Foster, Fangchen Liu, Federico Ceola, Fei Xia, Feiyu Zhao, Felipe Vieira Frujeri, Freek Stulp, Gaoyue Zhou, Gaurav S. Sukhatme, Gautam Salhotra, Ge Yan, Gilbert Feng, Giulio Schiavi, Glen Berseth, Gregory Kahn, Guangwen Yang, Guanzhi Wang, Hao Su, Hao-Shu Fang, Haochen Shi, Henghui Bao, Heni Ben Amor, Henrik I Christensen, Hiroki Furuta, Homer Walke, Hongjie Fang, Huy Ha, Igor Mordatch, Ilija Radosavovic, Isabel Leal, Jacky Liang, Jad Abou-Chakra, Jaehyung Kim, Jaimyn Drake, Jan Peters, Jan Schneider, Jasmine Hsu, Jeannette Bohg, Jeffrey Bingham, Jeffrey Wu, Jensen Gao, Jiaheng Hu, Jiajun Wu, Jialin Wu, Jiankai Sun, Jianlan Luo, Jiayuan Gu, Jie Tan, Jihoon Oh, Jimmy Wu, Jingpei Lu, Jingyun Yang, Jitendra Malik, João Silvério, Joey Hejna, Jonathan Booher, Jonathan Tompson, Jonathan Yang, Jordi Salvador, Joseph J. Lim,

Junhyek Han, Kaiyuan Wang, Kanishka Rao, Karl Pertsch, Karol Hausman, Keegan Go, Keerthana Gopalakrishnan, Ken Goldberg, Kendra Byrne, Kenneth Oslund, Kento Kawaharazuka, Kevin Black, Kevin Lin, Kevin Zhang, Kiana Ehsani, Kiran Lekkala, Kirsty Ellis, Krishan Rana, Krishnan Srinivasan, Kuan Fang, Kunal Pratap Singh, Kuo-Hao Zeng, Kyle Hatch, Kyle Hsu, Laurent Itti, Lawrence Yunliang Chen, Lerrel Pinto, Li Fei-Fei, Liam Tan, Linxi "Jim" Fan, Lionel Ott, Lisa Lee, Luca Weihs, Magnum Chen, Marion Lepert, Marius Memmel, Masayoshi Tomizuka, Masha Itkina, Mateo Guaman Castro, Max Spero, Maximilian Du, Michael Ahn, Michael C. Yip, Mingtong Zhang, Mingyu Ding, Minho Heo, Mohan Kumar Srirama, Mohit Sharma, Moo Jin Kim, Naoaki Kanazawa, Nicklas Hansen, Nicolas Heess, Nikhil J Joshi, Niko Suenderhauf, Ning Liu, Norman Di Palo, Nur Muhammad Mahi Shafiullah, Oier Mees, Oliver Kroemer, Osbert Bastani, Pannag R Sanketi, Patrick "Tree" Miller, Patrick Yin, Paul Wohlhart, Peng Xu, Peter David Fagan, Peter Mitrano, Pierre Sermanet, Pieter Abbeel, Priya Sundaresan, Qiuyu Chen, Quan Vuong, Rafael Rafailov, Ran Tian, Ria Doshi, Roberto Mart'in-Mart'in, Rohan Baijal, Rosario Scalise, Rose Hendrix, Roy Lin, Runjia Qian, Ruohan Zhang, Russell Mendonca, Rutav Shah, Ryan Hoque, Ryan Julian, Samuel Bustamante, Sean Kirmani, Sergey Levine, Shan Lin, Sherry Moore, Shikhar Bahl, Shivin Dass, Shubham Sonawani, Shuran Song, Sichun Xu, Siddhant Haldar, Siddharth Karamcheti, Simeon Adebola, Simon Guist, Soroush Nasiriany, Stefan Schaal, Stefan Welker, Stephen Tian, Subramanian Ramamoorthy, Sudeep Dasari, Suneel Belkhale, Sungjae Park, Suraj Nair, Suvir Mirchandani, Takayuki Osa, Tanmay Gupta, Tatsuya Harada, Tatsuya Matsushima, Ted Xiao, Thomas Kollar, Tianhe Yu, Tianli Ding, Todor Davchev, Tony Z. Zhao, Travis Armstrong, Trevor Darrell, Trinity Chung, Vidhi Jain, Vincent Vanhoucke, Wei Zhan, Wenxuan Zhou, Wolfram Burgard, Xi Chen, Xiangyu Chen, Xiaolong Wang, Xinghao Zhu, Xinyang Geng, Xiyuan Liu, Xu Liangwei, Xuanlin Li, Yansong Pang, Yao Lu, Yecheng Jason Ma, Yejin Kim, Yevgen Chebotar, Yifan Zhou, Yifeng Zhu, Yilin Wu, Ying Xu, Yixuan Wang, Yonatan Bisk, Yongqiang Dou, Yoonyoung Cho, Youngwoon Lee, Yuchen Cui, Yue Cao, Yueh-Hua Wu, Yujin Tang, Yuke Zhu, Yunchu Zhang, Yunfan Jiang, Yunshuang Li, Yunzhu Li, Yusuke Iwasawa, Yutaka Matsuo, Zehan Ma, Zhuo Xu, Zichen Jeff Cui, Zichen Zhang, Zipeng Fu, and Zipeng Lin. Open X-Embodiment: Robotic learning datasets and RT-X models. https://arxiv.org/abs/2310.08864, 2023.

Hanchen Cui and Yang Gao. A universal world model learned from large scale and diverse videos. In *NeurIPS 2023 Foundation Models for Decision Making Workshop*, 2023.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. The epic-kitchens dataset: Collection, challenges and baselines. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):4125–4141, 2020.

Hao-Shu Fang, Hongjie Fang, Zhenyu Tang, Jirong Liu, Junbo Wang, Haoyi Zhu, and Cewu Lu. Rh20t: A robotic dataset for learning diverse skills in one-shot. In *RSS 2023 Workshop on Learning for Task and Motion Planning*, 2023.

Shenyuan Gao, Siyuan Zhou, Yilun Du, Jun Zhang, and Chuang Gan. Adaworld: Learning adaptable world models with latent actions. In *International Conference on Machine Learning (ICML)*, 2025.

Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzyńska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, Florian Hoppe, Christian Thurau, Ingo Bax, and Roland Memisevic. The "something something" video database for learning and evaluating visual common sense, 2017. URL https://arxiv.org/abs/1706.04261.

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, Miguel Martin, Tushar Nagarajan, Ilija Radosavovic, Santhosh Kumar Ramakrishnan, Fiona Ryan, Jayant Sharma, Michael Wray, Mengmeng Xu, Eric Zhongcong Xu, Chen Zhao, Siddhant Bansal, Dhruv Batra, Vincent Cartillier, Sean Crane, Tien Do, Morrie Doulaty, Akshay Erapalli, Christoph Feichtenhofer, Adriano Fragomeni, Qichen Fu, Christian Fuegen, Abrham Gebreselasie, Cristina Gonzalez, James Hillis, Xuhua Huang, Yifei Huang, Wenqi Jia, Weslie Khoo, Jachym Kolar, Satwik Kottur, Anurag Kumar, Federico Landini, Chao Li, Yanghao Li, Zhenqiang Li, Karttikeya Mangalam, Raghava Modhugu, Jonathan Munro, Tullie Murrell, Takumi Nishiyasu, Will Price, Paola Ruiz Puentes, Merey Ramazanova, Leda Sari, Kiran Somasundaram, Audrey Southerland, Yusuke Sugano, Ruijie

Tao, Minh Vo, Yuchen Wang, Xindi Wu, Takuma Yagi, Yunyi Zhu, Pablo Arbelaez, David Crandall, Dima Damen, Giovanni Maria Farinella, Bernard Ghanem, Vamsi Krishna Ithapu, C. V. Jawahar, Hanbyul Joo, Kris Kitani, Haizhou Li, Richard Newcombe, Aude Oliva, Hyun Soo Park, James M. Rehg, Yoichi Sato, Jianbo Shi, Mike Zheng Shou, Antonio Torralba, Lorenzo Torresani, Mingfei Yan, and Jitendra Malik. Ego4d: Around the World in 3,000 Hours of Egocentric Video. In *IEEE/CVF Computer Vision and Pattern Recognition (CVPR)*, 2022.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

Danijar Hafner, Timothy P Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=0oabwyZbOu.

Haoran He, Yang Zhang, Liang Lin, Zhongwen Xu, and Ling Pan. Pre-trained video generative models as world simulators. *arXiv preprint arXiv: 2502.07825*, 2025.

Siqiao Huang, Jialong Wu, Qixing Zhou, Shangchen Miao, and Mingsheng Long. Vid2world: Crafting video diffusion models to interactive world models. *arXiv preprint arXiv: 2505.14357*, 2025a.

Siyuan Huang, Liliang Chen, Pengfei Zhou, Shengcong Chen, Zhengkai Jiang, Yue Hu, Yue Liao, Peng Gao, Hongsheng Li, Maoqing Yao, et al. Enerverse: Envisioning embodied future space for robotics manipulation. *arXiv preprint arXiv:2501.01895*, 2025b.

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

Harini Kannan, Danijar Hafner, Chelsea Finn, and Dumitru Erhan. Robodesk: A multi-task reinforcement learning benchmark. https://github.com/google-research/robodesk, 2021.

Yiming Li, Ziang Cao, Andrew Liang, Benjamin Liang, Luoyao Chen, Hang Zhao, and Chen Feng. Egocentric prediction of action target in 3d. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022.

Yin Li, Miao Liu, and James M Rehg. In the eye of beholder: Joint learning of gaze and actions in first person video. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 619–635, 2018.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *arXiv preprint arXiv:2306.03310*, 2023.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022a.

Yunze Liu, Yun Liu, Che Jiang, Kangbo Lyu, Weikang Wan, Hao Shen, Boqiang Liang, Zhoujie Fu, He Wang, and Li Yi. Hoi4d: A 4d egocentric dataset for category-level human-object interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21013–21022, June 2022b.

NVIDIA, :, Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi "Jim" Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, Joel Jang, Zhenyu Jiang, Jan Kautz, Kaushil Kundalia, Lawrence Lao, Zhiqi Li, Zongyu Lin, Kevin Lin, Guilin Liu, Edith Llontop, Loic Magne, Ajay Mandlekar, Avnish Narayan, Soroush Nasiriany, Scott Reed, You Liang Tan, Guanzhi Wang, Zu Wang, Jing Wang, Qi Wang, Jiannan Xiang, Yuqi Xie, Yinzhen Xu, Zhenjia Xu, Seonghyeon Ye, Zhiding Yu, Ao Zhang, Hao Zhang, Yizhou Zhao, Ruijie Zheng, and Yuke Zhu. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv: 2503.14734*, 2025.

OpenAI. Sora: Creating video from text. https://openai.com/sora, 2024. Accessed: 2025-09-18.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023.

Marc Rigter, Tarun Gupta, Agrin Hilmkil, and Chao Ma. AVID: Adapting video diffusion models to world models. In *Reinforcement Learning Conference*, 2025. URL https://openreview.net/forum?id=C18kcGeqAW.

Dominik Schmidt and Minqi Jiang. Learning to act without actions. *International Conference on Learning Representations*, 2023. doi: 10.48550/arXiv.2312.10812.

Richard S Sutton. Integrated architecture for learning, planning, and reacting based on approximating dynamic programming. In *Proceedings of the seventh international conference (1990) on Machine learning*, pp. 216–224, 1990.

Stephen Tian, Chelsea Finn, and Jiajun Wu. A control-centric benchmark for video prediction. *arXiv preprint arXiv:2304.13723*, 2023.

Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Jikai Wang, Qifan Zhang, Yu-Wei Chao, Bowen Wen, Xiaohu Guo, and Yu Xiang. Ho-cap: A capture system and dataset for 3d reconstruction and pose tracking of hand-object interaction, 2024a. URL https://arxiv.org/abs/2406.06843.

Xin Wang, Taein Kwon, Mahdi Rad, Bowen Pan, Ishani Chakraborty, Sean Andrist, Dan Bohus, Ashley Feniello, Bugra Tekin, Felipe Vieira Frujeri, Neel Joshi, and Marc Pollefeys. Holoassist: an egocentric human interaction dataset for interactive ai assistants in the real world. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20270–20281, October 2023.

Yucen Wang, Shenghua Wan, Le Gan, Shuai Feng, and De-Chuan Zhan. Ad3: Implicit action is the key for world models to distinguish the diverse visual distractors. *International Conference on Machine Learning*, 2024b. doi: 10.48550/arXiv.2403.09976.

Mingxing Xu, Wenrui Dai, Chunmiao Liu, Xing Gao, Weiyao Lin, Guo-Jun Qi, and Hongkai Xiong. Spatial-temporal transformer networks for traffic flow forecasting. *arXiv preprint arXiv:2001.02908*, 2020.

Seonghyeon Ye, Joel Jang, Byeongguk Jeon, Se June Joo, Jianwei Yang, Baolin Peng, Ajay Mandlekar, Reuben Tan, Yu-Wei Chao, Bill Yuchen Lin, Lars Liden, Kimin Lee, Jianfeng Gao, Luke Zettlemoyer, Dieter Fox, and Minjoon Seo. Latent action pretraining from videos. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=VYOe2eBQeh.

Weirui Ye, Yunsheng Zhang, Pieter Abbeel, and Yang Gao. Become a proficient player with limited data through watching pure videos. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Sy-o2N0hF4f.

Lixuan Zhang, Meina Kan, Shiguang Shan, and Xilin Chen. Prelar: World model pre-training with learnable action representation. In *European Conference on Computer Vision*, pp. 185–201. Springer, 2024.

Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all. *arXiv preprint arXiv: 2412.20404*, 2024.

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.

LARGE LANGUAGE MODELS (LLMS) USAGE

We used large language models (LLMs) solely as a writing and code-assistance tool, for tasks such as polishing text and providing autocomplete suggestions in code. The LLMs did not contribute to the research ideation, experimental design, data analysis, or interpretation of results. All scientific content, results, and conclusions are the original work of the authors.

A DATASET

We mainly focus on learning a latent action model and a world model for robotic manipulation that is adaptable to diverse downstream embodiments and action spaces. The data mixture for CoLA-World training is composed of both robot videos and human manipulation videos. For robot data, we primarily use Open X-Embodiment (OXE) (Collaboration et al., 2023) mixture and the AgiBot (AgiBot-World-Contributors et al., 2025) dataset. For human videos, we curate a comprehensive collection from nine prominent datasets, including Something-Something V2 (Goyal et al., 2017), RH20T (Fang et al., 2023), Ego4D (Grauman et al., 2022), EgoPAT3D (Li et al., 2022), EGTEA Gaze+ (Li et al., 2018), HOI4D (Liu et al., 2022b), EPIC-KITCHENS (Damen et al., 2020), HO-Cap (Wang et al., 2024a) and HoloAssist (Wang et al., 2023). The final data mixture consists of approximately 30% OXE, 20% AgiBot, and 50% human video data.

B IMPLEMENTATION DETAILS

Our two-stage training baseline involves training a LAM consisting of an IDM and an FDM, as well as a VQ quantizer to bottleneck the latent action space. Then the latent actions are inferred from the video using the frozen IDM and quantizer, used to finetune a pre-trained OpenSora video generation model into a world model, while the FDM is discarded. The joint training paradigm trains the LAM (i.e. the IDM and the VQ quantizer) and the OpenSora world model simultaneously, detaching the gradients of the world model's weights when executing warm-up. For fair comparison, the architectures of the IDM, the quantizer and the world model as well as the action conditioning moodules of the two paradigms are totally the same. We then elaborate each of the mentioned components above.

B.1 IDM, FDM AND THE QUANTIZER

The IDM is implemented as an 12-layer ST-Transformer (Xu et al., 2020). Each block has a hidden dimension of 768 and 12 attention heads. The FDM is implemented as an 12-layer spatial Transformer with the same number of hidden dimension and attention heads as the IDM. Between the IDM and FDM, we apply vector quantization (Van Den Oord et al., 2017) to produce latent actions, which is composed of two 32-dimensional action tokens chosen from the codebook. The codebook contains 32 entries, yielding a total number of 1024 different latent action choices. The IDM takes an $T \times 224 \times 224 \times 3$ video clip as input, first patchified with a patch size of 14 and then processed by the ST-Transformer to predict $T - 1$ latent actions. The FDM concatenates the image patches and the predicted latent action tokens, using the spatial transformer to produce pixel decoding results of the next frames. The IDM and FDM both have about 0.12 B parameters.

B.2 WORLD MODEL BASED ON THE PRETRINED OPENSORA MODEL

We adopt the pre-trained OpenSora model as the backbone of the world model. We use the v1.2 release with about 1.2 B parameters. As mentioned in Section 3.3, we add an extra from-scratch module for conditioning the video generation of OpenSora on the extracted latent actions, including 6 self-attention blocks to process the latent action sequence and an MLP to get the final AdaLN parameters of the latent actions, which are then fused with original diffusion timestep AdaLN parameters and modulate the attention results in each OpenSora DiT block. We initialize the weights in the action attention blocks as zero, to ensure a steady training at the beginning. Similar AdaLN-style action conditioning method is also explored in previous work (Zhu et al., 2024; He et al., 2025). However, their action inputs are fixed and not learnable, while our latent actions and conditioning layers are dynamically refined by the world model's own objective, which sets our method apart.

These newly introduced from-scratch modules to the OpenSora have about 74M parameters. The original layers in OpenSora for processing the texts, as well as the cross attention layers for fusing visual and text modalities, are discarded. Then there the about 0.93 B learnable parameters in the OpenSora, including the newly added action conditioning modules. Moreover, the original temporal transformer blocks in the OpenSora Dit are not causal, and we add causal masks in them to prevent future information influencing the past, which is unfavorable in dynamics modeling.

During training, the OpenSora WM takes in 256-resolution videos and the extracted latent action sequence, adding noise to the ground-truth videos and forwarding them through the DiT to predict the velocity vector, and building the prediction loss in the context of rectified flow. We use a timestep-wise classifier-free guidance, where during training we randomly mask the action condition as zero in a probability of 0.1 at each timestep of the sequence, and apply a guidance scale of 4.0 for sampling during inference. The number of denoising timestep is 10 in inference.

B.3 Training deatils

Latent Action Training of the two-stage paradigm After FDM producing pixel reconstruction results, we simply build the MSE loss between the reconstruction and the ground-truth "next frame" observation, in a teacher-forcing manner, rather than multi-step auto-regression. The vq loss and commitment loss introduced by the vq technique are also included to update the IDM and the codebook, and their loss weights are 1.0.

World Model Training of the two-stage paradigm As mentioned above, the OpenSora world model builds the flow matching loss using the input videos and the detached latent actions and update the OpenSora model, as well as the action conditioning modules.

Training of CoLA-World paradigm The OpenSora world model now builds the flow matching loss using the input videos and the learnable latent actions. The gradients then backpropagate throughout the whole system. The IDM, VQ quantizer and the action conditioning modules introduced in the OpenSora will be updated, while the pretrained weights of the original OpenSora model will only be updated after warm-up phase. The IDM and VQ quantizer will also receive gradients from the vq loss and commitment loss both during warm-up and end-to-end phase, similar to the latent action training in the two-stage paradigm.

Other training protocols. To ensure fair comparison, both training paradigms use a learning rate of 7.5e-5, a batch size of 128, and a 2000-step linear warmup schedule for the learning rate. When the LAM model is updating (LAM training of 2-stage paradigm, and all of the joint training paradigm), we use random crop to the video clips as a data augmentation trick to improve performance, while when the LAM is fixed, we do not use the augmentation and direct use the IDM to extract the latent actions from the original video.

C Evaluation Details

C.1 Evaluation Setup

For linear probing task and all the video prediction tasks, we train the prober head (the LAM and the world model) on the training split of the given dataset mixture, and validate on the valid split. For example, for linear probing on out-of-distribution LIBERO dataset, in fact the LAM is previously trained on the whole training data, and the prober head is now trained on the training split of the unseen LIBERO dataset. Then, we test the performance of the LAM and the prober by probing the loss on the valid split of the LIBERO dataset, and record the results. For all the probing tasks, we train the prober head for 1K steps with a batch size 64 on 8 gpus (512K samples in all), and validate on 20K test samples. For all the video prediction tasks, we evaluate on a fixed test dataset for each data mixture, consisting of 240 video clips on each gpu, and the performance is averaged.

C.2 Real Action Adaptation

When adapting the trained world model to a downstream real action space, we first train the adapter predicting the GT-LAM vq code indices from the real actions using a 2-layer MLP. This takes 1K

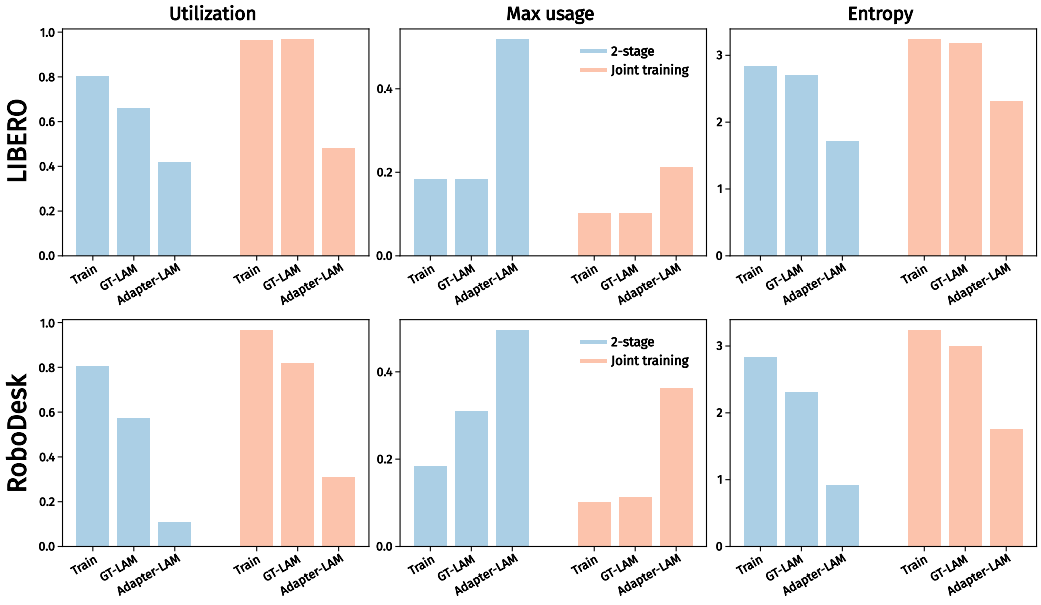

Figure 5: Codebook metrics in different training and adaptation stages. All subplots share the same legend, shown only in the middle panel for clarity.

steps training with a batch size of 64. We then finetune the world model on downstream dataset using Gt-LAM for 3K steps with a batch size of 16.

C.3 Visual planning on VP^{2}enchmark

We test the learned world model's utility in control on RoboDesk environment using the evaluation protocol from VP^2 benchmark. Each task of the RoboDesk environment in VP^2 benchmark is specified by 30 pairs of initial observation and goal observation. When testing on one task, every time we sample such a pair and the agent needs to use the world model to plan the trajectory starting at the initial state towards the goal. The reward function is also provided by VP^2, defined as the weighted sum of the MSE loss between the predicted video and the goal observation, with a pretrained binary classifier's predicted logit on the current task. The classifier's weights are also provided by the benchmark. Finally, the task success rate is the ratio of success trajectories in these 30 runs. Moreover, VP^2 offers trajectory data on RoboDesk, and the experiments of world model downstream adaptation on RoboDesk in Section 4.4 is conducted by training the adapter and finetuning the world model on these data.

D Additional Results

D.1 Analysis of Codebook Dynamics in Downstream Adaptation

To provide deeper quantitative insight into the mechanisms behind our paradigm's superior downstream real-action-adaptation performance over two-stage method, we analyze the metrics of the VQ codebook. For both CoLA-World and the Two-Stage baseline, we compare three distinct latent action distributions on the LIBERO and RoboDesk datasets:

(a) Training Distribution: The latent action distribution in our general training.

(b) GT-LAM Fine-tuning Distribution: The ground-truth latent action distribution inferred by the frozen LAM encoder from the downstream task videos, used for fine-tuning the world model.

(c) Adapter-LAM Inference Distribution: The latent action distribution produced by the trained adapter when translating the downstream task's real actions.

The results, visualized in Figure 5, reveal a stark contrast in how the two paradigms adapt their latent action space.

As shown in the bar charts, the two-stage method exhibits a dramatic representational collapse when adapting to the downstream tasks' real actions. While the codebook utilization and entropy are reasonable during pre-training (a), they decrease when the model is fine-tuned on the narrower distribution of the downstream GT-LAM (b). Most critically, when the adapter is used for inference (c), the codebook metrics degenerate severely and tend to collapse: codebook utilization plummets to nearly 10% on RoboDesk, with the max_usage metric spiking to approximately 0.5 on both LIBERO and RoboDesk. This indicates that the adapter has found a "lazy shortcut" by mapping the vast majority of real actions to a single, all-purpose latent code. This is a direct cause of the model's low performance and its inability to handle the full complexity of the control task.

In contrast, the overall codebook usage is relatively healthy in our CoLA-World paradigm under the Adapter-LAM setting. The entropy remains high and the max_usage stays at a relatively low level compared to the two-stage baseline. This provides direct, quantitative evidence that the co-evolutionary process has forged a more robust and flexible latent action space for downstream adaptation and generalization. The constant, supervisory feedback from the powerful world model tutor prevents the LAM from taking degenerative shortcuts, compelling them to learn a richer, more meaningful representations. This preserved diversity of the codebook is a cornerstone of our system's adaptation performance and its ability to robustly generalize.

To conclude, and in conjunction with our analysis in Section 4.4, our joint training paradigm's success in downstream adaptation stems from co-evolution forging an intrinsically consistent and deeply coupled system, which manifests in the dual advantages of a collapse-resistant latent action space and a world model that robustly utilizes it.

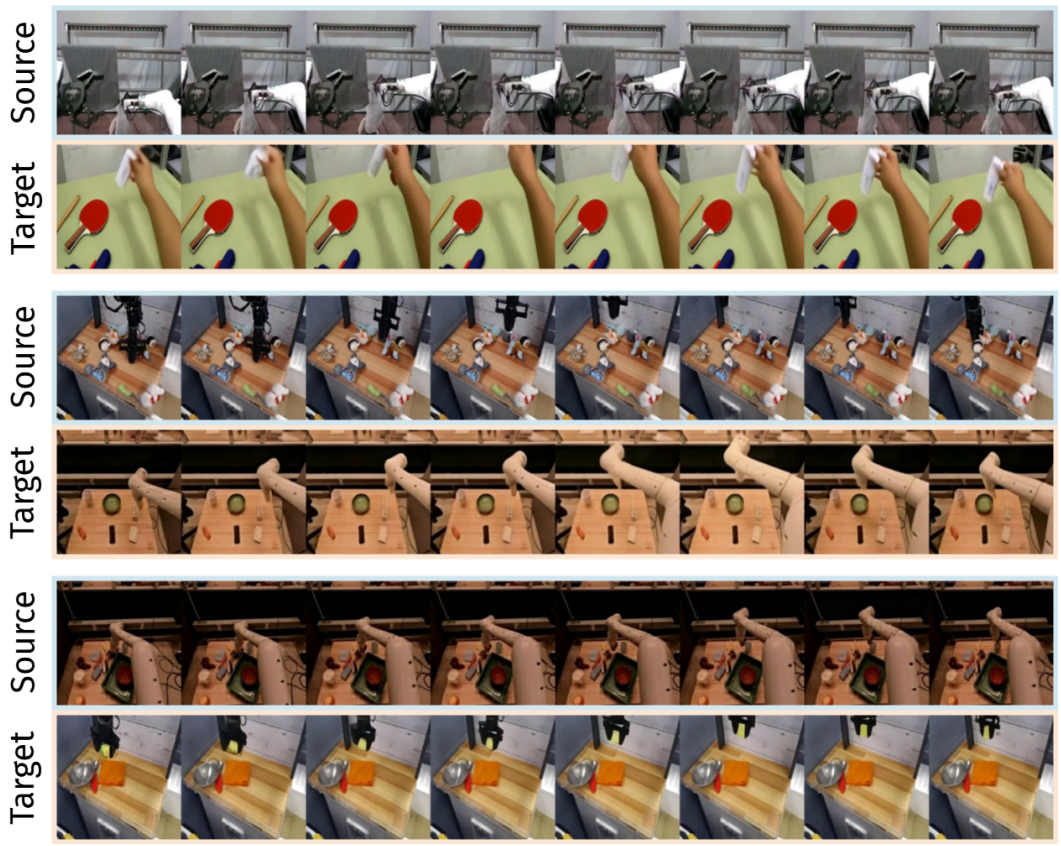

Figure 6: Action transfer results. The first image of the generated video comes from a different dataset from the source video.

D.2 Action Transfer results

Here we provide action transfer results in Figure 6, where our learned LAM in CoLA-World extracts the latent actions from the source video, and the world model generates the video from an initial image, taking these latent actions as conditions. For each video pair below, the top video is the source video, while the bottom one is the generated action-transfer video. We notice that the generated videos show a strong resemblance in semantic meaning to the source videos. To avoid too large PDF file, we provide additional qualitative results for action transfer videos in our anonymous repository.