PRE-TRAINED VIDEO GENERATIVE MODELS AS WORLD SIMULATORS

Haoran He[1], **Yang Zhang**[2], **Liang Lin**[3], **Zhongwen Xu**[4], **Ling Pan**[1†]
[1]Hong Kong University of Science and Technology [2]Tsinghua University
[3] Sun Yat-sen University [4] Tencent AI Lab

ABSTRACT

Video generative models pre-trained on large-scale internet datasets have achieved remarkable success, excelling at producing realistic synthetic videos. However, they often generate clips based on static prompts (e.g., text or images), limiting their ability to model interactive and dynamic scenarios. In this paper, we propose **D**ynamic **W**orld **S**imulation (DWS), a novel approach to transform pre-trained video generative models into controllable world simulators capable of executing specified action trajectories. To achieve precise alignment between conditioned actions and generated visual changes, we introduce a lightweight, universal action-conditioned module that seamlessly integrates into any existing model. Instead of focusing on complex visual details, we demonstrate that consistent dynamic transition modeling is the key to building powerful world simulators. Building upon this insight, we further introduce a motion-reinforced loss that enhances action controllability by compelling the model to capture dynamic changes more effectively. Experiments demonstrate that DWS can be versatilely applied to both diffusion and autoregressive transformer models, achieving significant improvements in generating action-controllable, dynamically consistent videos across games and robotics domains. Moreover, to facilitate the applications of the learned world simulator in downstream tasks such as model-based reinforcement learning, we propose prioritized imagination to improve sample efficiency, demonstrating competitive performance compared with state-of-the-art methods.

1 INTRODUCTION

The field of video generation has experienced remarkable progress in recent years, with models such as Brooks et al. (2024); Zheng et al. (2024); Polyak et al. (2024); Yang et al. (2024c); Sharma et al. (2024) demonstrating an exceptional ability to generate high-fidelity and temporally consistent videos conditioned on various inputs, most notably text and initial frames. However, these models are limited to support interactive simulation scenarios, as they are trained for one-shot generation with static prompts, lacking frame-level interactivity and frame-to-frame dynamic modeling. To fill this gap, the community is increasingly focusing on building action-conditioned video models Yang et al. (2023); Bruce et al. (2024); Xiang et al. (2024); Wu et al. (2024); Valevski et al. (2024); Decart et al. (2024); Che et al. (2025); Yang et al. (2024a).

These action-conditioned models effectively act as interactive environment simulators ("world models" or "world simulators"), which leverage advanced transformers or diffusion model architectures to predict future visual outcomes based on the agent's actions. Their goal is to encapsulate an understanding of the underlying dynamic transitions and commonsense knowledge about how the world works, enabling action-driven imagination analogous to the human cognition process. These world models open exciting possibilities, particularly in model-based reinforcement learning (MBRL), where agents can learn new skills more efficiently by interacting with world models, avoiding the risks and costs that arise from real-world trials.

In this work, we review recent advances in interactive world simulators, highlighting key challenges that currently limit their broader adoption. (i) These models often require vast computational resources for training from scratch. For example, Genie Bruce et al. (2024) required 125k training steps on 256 TPUv5p cores (roughly equivalent to 226 NVIDIA A100 GPUs) to learn a relatively simple

†Correspondence to: Ling Pan (lingpan@ust.hk).

Platformer game simulator. Similarly, GameNGen Valevski et al. (2024) consumed 700k steps with 128 TPU-v5e cores. (ii) While fine-tuning a pre-trained video generative model offers a more efficient alternative, existing approaches (Rigter et al., 2024; Yu et al., 2025) are inherently architecture-dependent. This poses challenges to adapting these methods across different model architectures to benefit from rapid advances in video generation architectures, as each new model architecture requires substantial engineering efforts for adaptation. (iii) Unlike general video generation tasks, action-conditioned world simulators require precise capture of fine-grained dynamic changes (Zhu et al., 2024; Yang et al., 2023), which requires frame-level action alignment. This requirement is crucial for applications in model-based reinforcement learning, where capturing frame-to-frame dynamic/motion changes takes precedence over modeling static visual elements (e.g., background, object details).

To address the aforementioned challenges, we propose a novel framework, **D**ynamic **W**orld **S**imulation (DWS), which is a unified, architecture-agnostic approach for efficiently converting pre-trained video generative models into world simulators. By leveraging pre-trained priors learned from internet-scale datasets, the fine-tuned world simulators can demonstrate a basic understanding of physical rules and commonsense knowledge. However, they are not inherently equipped for interactive simulation and lack the key mechanisms for precise frame-level action conditioning. DWS introduces a minimalist yet powerful add-on action-conditioned module that improves frame-level action awareness while maintaining architectural flexibility. This module comprises just two linear layers and can be integrated into any network architecture through carefully designed scale and shift operations. It also strengthens the alignment between predicted visual changes and conditioned actions. Furthermore, we observe that traditional supervised learning loss functions lead video models to focus uniformly across all visual elements and complex details, including static backgrounds and irrelevant details, compromising their ability to capture the frame-to-frame dynamic/motion changes crucial for world simulator construction. DWS presents a motion-reinforced loss to address the dynamic modeling challenge, a simple yet effective method to explicitly prioritize the modeling of inter-frame changes during training, resulting in significantly improved temporal consistency and more reliable dynamic predictions. As shown in Fig. 1, the learned world simulators by DWS can interact with diverse policies across different domains while maintaining accurate dynamic prediction and action responsiveness. Finally, to enhance the practical utility of world simulators in model-based reinforcement learning, we introduce prioritized imagination, a novel sampling strategy that focuses on the most informative transitions rather than wasting computational resources on well-understood state transitions, leading to improved sample efficiency during agent-world model interactions.

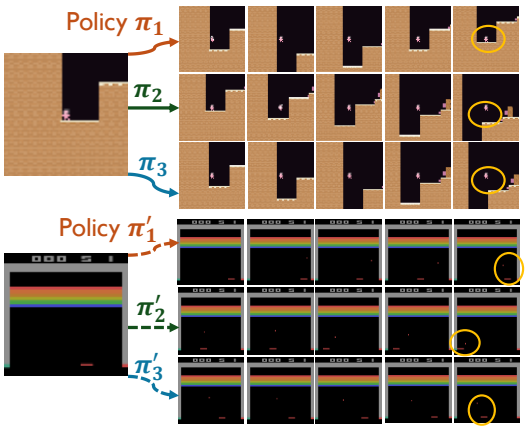


Figure 1: Our fine-tuned video generative model can serve as an effective world simulator for interacting with different policies and generating diverse trajectories. Different policies can lead to different terminal states, which are highlighted with yellow circles.

We summarize the contributions of this paper as follows: (i) We introduce DWS, a novel and architecture-agnostic framework that effectively converts pre-trained video generative models to world simulators with low training costs, leveraging the pre-trained prior knowledge for physics grounding. (ii) We introduce two simple yet effective techniques: a lightweight action-conditioned module that enables precise frame-level control and improve action-following ability, and a motion-reinforced training that redirects model attention from static visual details to action-induced dynamic changes for improving temporal, dynamic consistency. (iii) We advance the practical utility of world simulators in model-based reinforcement learning through prioritized imagination, which improves sample efficiency and policy performance when applying the trained world simulators to downstream model-based RL. (iv) Through comprehensive evaluation across challenging game and robotics tasks, we demonstrate that DWS significantly improves the quality and dynamic consistency of generated action-conditioned videos. Our DWS-trained world simulators with prioritized imagination also enable

more efficient and effective learning compared to previous state-of-the-art MBRL approaches (Hafner et al., 2023; Alonso et al., 2024).

2 Related Work

Video World Models. With the development of internet-scale datasets (Bain et al., 2021; Chen et al., 2024) and advanced model architecture (Peebles & Xie, 2023; Brooks et al., 2024), significant progress has been made in realistic video generation conditioned on text descriptions and initial frames (Blattmann et al., 2023; Lin et al., 2024; Ma et al., 2024; Yang et al., 2024c; Zheng et al., 2024). Building upon these foundations, current research has increasingly focused on action-controllable video generation, aiming to develop generalist world simulators (Xiang et al., 2024; Yang et al., 2023; Bruce et al., 2024; Feng et al., 2024; Valevski et al., 2024; Parker-Holder et al., 2024; Zhu et al., 2024; Che et al., 2025) that can effectively model both physical dynamics and action consequences. However, these models typically require training from scratch on large-scale datasets and involve millions (or billions) of parameters, resulting in substantial computational overhead and slow inference speed. In contrast, we propose to adapt publicly available pre-trained video generative models (Zheng et al., 2024; Wu et al., 2024) into action-driven world simulators. Our proposed fine-tuning approach, DWS, achieves efficient model adaptation while requiring minimal computational overhead, significantly reducing both training costs and inference latency. Concurrent works (Rigter et al., 2024; Yu et al., 2025) have also explored leveraging pre-trained models for action-conditioned video generation. However, their investigations are limited to diffusion-based models, and neither work validates the effectiveness of their approaches in facilitating downstream tasks such as model-based RL.

Model-Based RL Model-based RL aims to build world models in which the trial-and-errors can take place without real cost. With a sufficiently accurate world model, agents can develop imagination abilities, allowing them to simulate interactions and generate synthetic experience data. This simulated data can then be leveraged to learn optimal policies for diverse decision-making tasks, effectively reducing the need for real-world interactions. Sutton (1991) introduce the first general framework for model-based RL, highlighting the utility of an estimated dynamics model in facilitating the training of value functions and policies (Sutton & Barto, 1998). Recent years have witnessed remarkable progress in model-based methods for learning complex environmental dynamics, such as video games and visual control tasks, consistently outperforming their model-free counterparts. For example, built upon Recurrent State Space Models (RSSM) (Hafner et al., 2019), which explicitly decouple the deterministic and stochastic components of environmental dynamics, the Dreamer series has demonstrated impressive performance across diverse domains, including Atari games (Machado et al., 2018), DeepMind Control Suite (Tassa et al., 2018), and Minecraft. To address the limitation of RNNs in expressing complex patterns, recent works have explored leveraging transformer models for enhanced sequence modeling and long-term dependency capture (Micheli et al., 2023a; Robine et al., 2023; Zhang et al., 2023; 2024), and incorporating diffusion models to better represent multi-modal distributions in dynamic learning (Ding et al., 2024; Alonso et al., 2024). However, although these works also employ transformer or diffusion models for world model learning, they predominantly rely on training from scratch and fail to leverage pre-trained knowledge for enhanced dynamics understanding, making them overly task-specific and limiting their ability to generalize across diverse tasks. Furthermore, while existing methods treat all imagined samples with uniform importance during training, our proposed DWS introduces a novel prioritization mechanism that selectively focuses on significant samples, thereby improving sample efficiency.

3 Preliminaries

3.1 Problem Formulation

The conditional video generation framework can be adaptable to instantiate a world simulator (or a world model) (Yang et al., 2023). The world model takes in some action as input and produces the visual consequence of the action as output, which aims to simulate the environment. This environment can be represented as a Partially Observable Markov Decision Process (POMDP), encapsulated within the tuple $(\mathcal{S}, \mathcal{O}, \phi, \mathcal{A}, p, r, \gamma)$. Here, $\mathcal{S}$ is the state space, and $\mathcal{O}$ is the observation space which only provides incomplete information of $\mathcal{S}$. At each timestep t, the agent chooses an action a_t by following a policy $\pi : \mathcal{O} \to \Delta_{\mathcal{A}}$, the environment updates the state following the dynamics, $s_{t+1} \sim p(s_{t+1}|s_t, a + t)$, the next observation $o_{t+1} = \phi(s_{t+1})$ is received and a scalar reward r_t is computed as $R(s_t, a_t, s_{t+1})$. The goal of the agent is to learn a policy $\pi^* = \arg\max_\pi \mathbb{E}_{a_t \sim \pi}\left[\sum_{t=0}^{\infty} \gamma^t r_t\right]$ by maximizing the γ-discounted cumulative rewards.

A well-trained world model can replace the environment to interact with the agent, and thus benefit downstream policy learning by providing infinite experiences. Concretely, given a history observation o_{T_0}, at each timestep $t = T_0, \cdots, T-1$, the agent takes an action a_t based on its policy and previous imagined observations, and then the world model predicts the transition $p(o_{t+1}, r_{t+1}|o_t, a_t)$ to feedback the agent.

3.2 Pre-Trained Video Generative Models

By formulating learning world models for visual control as an interactive video generation problem, we can harness the widely available video data, which embeds broad knowledge that is generalizable across different domains (Yang et al., 2024b). Video data not only contains semantic visual details but also includes motion movements that capture the dynamic rules in the physical world. However, training such video world models on internet-scale video datasets from scratch is expensive and time-consuming. We propose to fine-tune pre-trained advanced video models to enable them to simulate interactions. Specifically, we adopt two different pre-trained video generative models as our base models, which are diffusion models, i.e., Open-Sora (Zheng et al., 2024), and autoregressive transformer models, i.e., iVideoGPT (Wu et al., 2024). Open-Sora is a kind of rectified flow-based diffusion model that is fully open-sourced and pre-trained on millions of internet videos. We consider using it because it requires only a few sampling steps, benefiting from flow-matching training. A recent work named iVideoGPT is an autoregressive transformer built upon LLaMA (Touvron et al., 2023) architecture. It compresses training data from different modalities (e.g., including visual outcomes, actions, and rewards) into a sequence of tokens for interactive video prediction.

4 Method

In this section, we first introduce our proposed action-conditioned module in §4.1, and the motion-reinforced loss for enhancing dynamic modeling in §4.2. After presenting methods for fine-tuning general video generative models into world simulators, we introduce the prioritized imagination technique for improving model-based reinforcement learning performance in §4.3.

4.1 Action-Conditioned Module

Recent video generative models have achieved significant success in generating realistic videos correlated with conditioned text prompts or initial frames. These text-to-video tasks operate with static prompts that globally describe the entire video without specifying what the next frames should be. This design paradigm, while effective for general video generation, presents fundamental challenges when adapting them as world models that aim to simulate action-rich interactions, where the conditions are frame-level, fine-grained action trajectories. To address this requirement and ensure each generated frame matches its corresponding action in the trajectory, we leverage an action-conditioned module that conditions the generation of each frame by its corresponding action individually. Unlike previous text-to-video models that compress the entire action trajectory into a single embedding, our approach, similar to IRASim (Zhu et al., 2024), implements a more granular action encoding mechanism. We introduce a lightweight add-on module, consisting of two linear layers within each transformer block, to encode individual actions separately. This design ensures that each frame's content is directly modulated by its corresponding action, rather than being guided by a global description, and leads to a direct correspondence between actions and generated frames.

Action Representation. A key challenge in adapting video generative models for action-based control lies in the representation of actions. In discretized action spaces, actions are typically represented as integer values, which lack the rich semantic context present in text prompts used in traditional text-to-video models. This semantic gap can limit the model's ability to interpret and respond to different actions effectively. To bridge this gap, we propose to represent the actions using language templates that depict the meanings of the actions. Specifically, given an action trajectory $y = \{a_t, a_{t+1}, \cdots, a_{t+H-1}\}$, where H is the horizon of the trajectory, we develop a mapping function ψ to translate abstract action integers into meaningful languages, i.e., $\psi : \mathcal{A} \to L$, where L is the language space. This mapping enables us to leverage the text encoder in pre-trained video generative models to obtain rich feature embeddings $c \in \mathbb{R}^{n_H \times n_d}$, where n_H and n_d represent the horizon and the dimension of each token respectively. For continuous action spaces, following Wu et al. (2024); Zhu et al. (2024), we use a trainable linear action embedder to directly generate feature embeddings c without language translation.

Frame-Level Condition. In the context of video generative models serving as world simulators, precise temporal control is important as each action should directly modulate the visual content of its

subsequent frame. To explicitly model and enhance this action-frame correspondence, we incorporate a frame-level action-conditioning module within each transformer block, drawing inspiration from IRASim (Zhu et al., 2024). While IRASim's implementation was limited to specific diffusion models with temporal-spatial transformer architectures, we significantly extend this concept by developing a versatile add-on module that generalizes across different model architectures, including both diffusion-based and transformer-based frameworks. Therefore, our design offers enhanced architectural flexibility and broader applicability. Our minimalist architecture, implemented with just two linear layers, enables lightweight integration and efficient fine-tuning with minimal computational overhead. Specifically, for each video embedding $x \in \mathbb{R}^{T\times C\times H\times W}$, we process them as follows before feeding them into the transformer block:

$$x^i = x^i + \text{FFN}(\text{LayerNorm}(x^i) \times (1 + \alpha^i) + \beta^i), \tag{1}$$

where α^i and β^i denote the scale and shift parameters for the i-th frame. They are regressed from the action embedding c^i. We illustrate our proposed module in Fig. 2, which can be seamlessly integrated into any network block (e.g., attention block).

4.2 Motion-Reinforced Loss

Traditional video generative models commonly employ the squared l_2 distance (for rectified flow (Liu et al., 2022; Zheng et al., 2024) in the continuous space) or cross-entropy loss (for next-token-prediction transformers (Vaswani et al., 2017; Wu et al., 2024) in discrete space) as their training objectives. While these training objectives have demonstrated effectiveness in general video generation tasks (Zheng et al., 2024; Lin et al., 2024; Brooks et al., 2024; Yan et al., 2021; Tian et al., 2024), they typically consider each pixel equally, which may compromise the model's ability to capture action-dependent state changes. Therefore, it is inefficient for them to function effectively as world simulators for RL agents, where accurate modeling of dynamic transitions is more crucial for learning than maintaining high-fidelity background details. This limitation arises because RL agents predominantly learn from action-induced state changes rather than static visual elements.

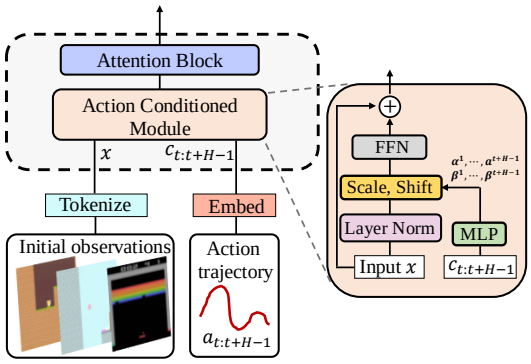


Figure 2: Illustration of the action-conditioned module, which can be incorporated with any type of transformer attention block.

To tackle this problem and enable more precise dynamic modeling, we introduce a new motion-reinforced loss to improve the action-following ability of video models. At each training step, we sample a random batch of ground-truth video embeddings $x = \{x^0, \cdots, x^k, \cdots, x^j, x^{H-1}\}$, where H denotes the horizon of the video clips. We then compute the differences $\omega = \cup_{i=0}^{H-1}\omega_i$ between consecutive frames, denoted as

$$\omega_{i+1} = c^{\text{Softmax}(|x_{i+1}-x_i|)}, \tag{2}$$

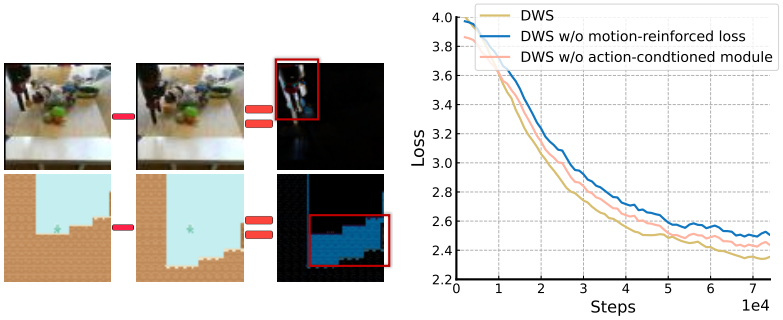
where $\omega_i \to [1, c]$, and we set $\omega_0 = 1$ for the initial frame x^0 since it serves as a conditioned frame. Here, c denotes a hyperparameter that modulates the motion-reinforced strength. After obtaining ω, we integrate it as pixel-wise weights into the supervised training loss. The resulting motion-reinforced loss function can be formulated as:

$$\mathcal{L}_{\text{motion}} = \mathcal{L}_{\text{prev}} * \omega, \tag{3}$$

where $\mathcal{L}_{\text{prev}}$ represents either the original MSE loss used in diffusion models, or the cross-entropy loss function in transformer-based architectures. We include more implementation details in Appendix B.2. Through this formulation, pixels that change across frames will have a greater impact on loss backpropagation. These pixels typically correspond to motion-related elements in videos, which undergo continuous changes, while the background, which remains stable, will have less influence during training. This mechanism inherently attenuates the impact of static background elements that contribute minimally to action-conditioned prediction.

By emphasizing dynamic transitions, our approach enhances the world model's capability to capture action-state causal relationships, thereby facilitating more effective policy learning in reinforcement learning contexts.

As illustrated in Figure 3(a), inter-frame differences predominantly correspond to motion-related and dynamic elements in videos, leading ω to assign higher weights to these pixels during the training



(a) frame-to-frame differences (b) SFT loss $\mathcal{L}_{\text{prev}}$ comparison

Figure 3: Motion-reinforced loss enhances the action-controllability of video generative models, helping capture dynamically changing contents.

process. Figure 3(b) presents the SFT loss ($\mathcal{L}_{\text{prev}}$) curves from fine-tuning iVideoGPT (Wu et al., 2024) on the BAIR dataset Ebert et al. (2017), comparing different variants of our proposed method. The empirical results demonstrate that both the motion-reinforced loss and the action-conditioned module are crucial components, as the absence of either component significantly degrades the model's performance in predicting action-conditioned videos with frame-to-frame dynamics.

Therefore, the video generative models fine-tuned by $\mathcal{L}_{\text{motion}}$ will focus more on the dynamic/motion prediction instead of complex visual details that are challenging to learn. Moreover, dynamic/motion consistency is more important than static background details for world simulators, since simulators are required to predict visual outcomes conditioned on actions.

4.3 Model-Based Reinforcement Learning

Given the video-based world simulators fine-tuned from pre-trained video generative models, one of the most promising applications is to utilize them as world models for policy learning in model-based reinforcement learning (MBRL). In MBRL, the agent optimizes a policy to maximize the cumulative rewards by interacting with the trained world models which improves sample efficiency.

To enable effective policy training in MBRL, the world model should predict both transition dynamics and rewards. We now complete our world model with a reward prediction model. Since estimating the reward is a scalar prediction problem, we introduce a separate model R_ψ consisting of linear layers, self-attention blocks, and cross-attention blocks to estimate the reward given past observations and actions. The RL agent involves an actor-critic network parameterized by a shared CNN backbone that branches into separate policy and value heads. Building upon the MBPO framework (Janner et al., 2019b; Wu et al., 2024), we augment the replay buffer with synthetic rollouts to train a standard actor-critic RL algorithm. We adopt PPO (Schulman et al., 2017) as our base algorithm and follow Huang et al. (2022) for implementation. We include more implementation details in Appendix B.3.

Prioritized Imagination. Imagination by a world model needs to start from initial observations, which are sampled from the experiences collected from the environment. These initial states serve as starting points for the world model to generate synthetic trajectories. Previous MBRL methods (Hafner et al., 2020a;b; 2023; Alonso et al., 2024; Micheli et al., 2023b) employ uniform sampling of initial observations for imagination. However, this strategy neglects the varying importance of different states for policy learning, leading to learning inefficiency. We highlight that imagined transitions originating from different initial observations exhibit substantial heterogeneity in their importance and task relevance for MBRL policy optimization. To better unlock the world simulation ability of fine-tuned video generative models, we propose a prioritized imagination method that selectively focuses on more valuable transitions. Our key insight is that initial observations leading to transitions with higher learning potential and learn-ability should be sampled more frequently. We maintain a buffer $\mathcal{B}$ to store observations encountered during the interaction with environments, and prioritize initial observations with high expected learning progress, which is measured by the magnitude of their TD loss. This prioritization mechanism ensures more efficient utilization of the world model by concentrating imagination resources on states that yield more substantial contributions to policy learning. The overall pseudo-code is presented in Algorithm 1.

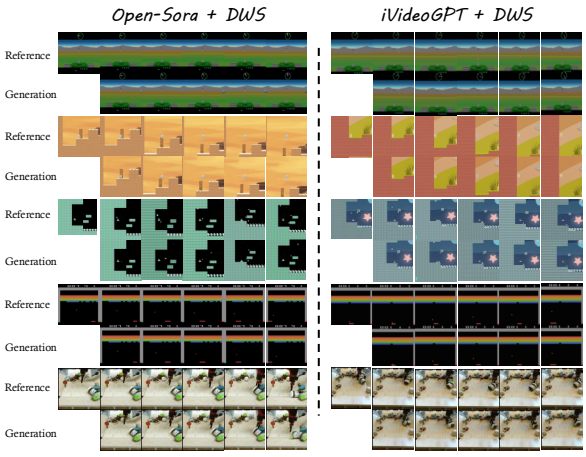


Figure 4: The qualitative results of generated videos for different domains, including games and robotics environments. Given initial observations and conditioned actions, we observe that Open-Sora and iVideoGPT fine-tuned by our proposed method significantly improve dynamic world modeling ability.

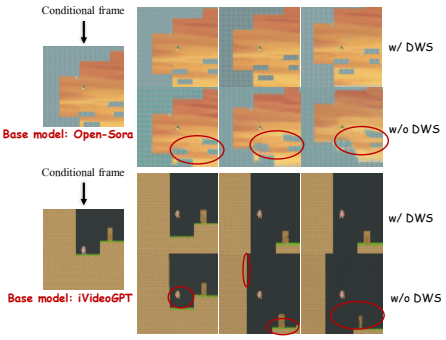


Figure 5: Quantitative comparison between pre-trained video generative models and fine-tuned models by our proposed methods. Base models without DWS can generate inconsistent pixels and fail to match the conditioned actions, as highlighted in red circles.

5 Experiments

In this section, we evaluate the video world model fine-tuned by our proposed method from two critical perspectives: (1) action simulation capability, assessed through the quality of action-conditioned video prediction, and (2) the effectiveness in both online and offline model-based reinforcement learning, quantified by the cumulative return across tasks.

5.1 Action-Conditioned Simulation

To evaluate the effectiveness of DWS in enhancing action-conditioned video prediction for world simulation, we conduct experiments using two architecturally distinct pre-trained video generative models: Open-Sora (Zheng et al., 2024), which employs a diffusion-based architecture, and iVideoGPT (Wu et al., 2024), which utilizes an autoregressive architecture. For computationally efficiency, we utilize a compressed version of Open-Sora with 12 layers (approximately 280M parameters) instead of the original 1.1B model. The details and our setup of these base models can be found in Appendix B.1. The evaluation is performed across three diverse datasets: the BAIR dataset (Ebert et al., 2017) featuring continuous action spaces, and both Procgen (Cobbe et al., 2020) and Atari (Bellemare et al., 2013; Machado et al., 2018) datasets incorporating discrete action spaces.

Experiment Setup. The BAIR robot pushing dataset which is about a robotic arm manipulating various objects consists of 43k training and 256 test videos, Following previous works (Yan et al., 2021; Gupta et al., 2022), we predict 15 frames from a single initial frame. In this benchmark, we compare against a variety of video prediction models, including diffusion (Zheng et al., 2024; Voleti et al., 2022), masked (Yu et al., 2023; Gupta et al., 2022), and autoregressive models (Wu et al., 2024; Yan et al., 2021). For the Procgen dataset, we evaluate DWS on two platformer games, i.e., namely Coinrun and Ninja. For the Atari dataset, we assess performance on two Atari games: Breakout and Battle Zone. We collect 1M transition steps for each game, encompassing RGB observations, actions, and rewards. The resolution is 64×64 for both BAIR and Procgen videos, while the resolution for Atari videos is 84×84. More details regarding dataset collection are provided in Appendix A.

Metrics. We adopt four widely-used metrics to measure the quality of predicted videos: across four metrics: 1): FVD (Unterthiner et al., 2018) quantifies the statistical similarity between reference and generated video distributions by computing the Fréchet distance between feature representations extracted from a pre-trained network (Carreira & Zisserman, 2017); 2) PSNR (Huynh-Thu & Ghanbari, 2008) measures the pixel-wise fidelity between reference and generated frames, which is computed as the logarithmic ratio between the maximum possible pixel value and the mean squared error; 3) SSIM (Huynh-Thu & Ghanbari, 2008) evaluates the structural information preservation in generated frames, which incorporates luminance, contrast, and structural comparisons; 4) LPIPS (Zhang et al.,

2018) leverages a pre-trained deep neural network (Simonyan & Zisserman, 2014) to assess image similarity by computing distances in deep feature space.

Table 1: Video prediction results on the BAIR robot pushing dataset. LPIPS and SSIM scores are scaled by 100 for convenient display.

BAIR Ebert et al. (2017)	FVD↓	PSNR↑	SSIM↑	LPIPS↓
VideoGPT Yan et al. (2021)	103.3	-	-	-
MaskViT Gupta et al. (2022)	93.7	-	-	-
FitVid Babaeizadeh et al. (2021)	93.6	-	-	-
MaskViT Gupta et al. (2022)	70.5	-	-	-
MCVD Voleti et al. (2022)	89.5	16.9	78.0	-
MAGVIT Yu et al. (2023)	62.0	19.3	78.7	12.3
Open-Sora (Zheng et al., 2024)	92.1	21.5	84.7	8.6
Open-Sora**+DWS**(Ours)	**81.3**	**22.4**	**87.8**	**6.2**
iVideoGPT Wu et al. (2024)	60.8	24.5	90.2	5.0
iVideoGPT**+DWS** (Ours)	**59.6**	**25.8**	**91.6**	**4.7**

Table 2: Video prediction results on Atari and Procgen game domains.

Domain	Method	FVD↓	PSNR↑	SSIM↑	LPIPS↓
Atari	Open-Sora (Zheng et al., 2024)	27.8	31.6	95.2	6.4
	Open-Sora**+DWS**(Ours)	**16.3**	**35.8**	**98.0**	**4.9**
	iVideoGPT Wu et al. (2024)	11.5	39.1	97.3	3.1
	iVideoGPT**+DWS** (Ours)	**9.1**	**43.4**	**98.2**	**1.2**
Procgen	Open-Sora (Zheng et al., 2024)	37.6	22.7	74.7	13.6
	Open-Sora**+DWS** (Ours)	**28.2**	**23.9**	**76.1**	**12.8**
	iVideoGPT Wu et al. (2024)	24.0	25.3	77.6	12.1
	iVideoGPT**+DWS** (Ours)	**21.9**	**26.2**	**80.4**	**10.3**

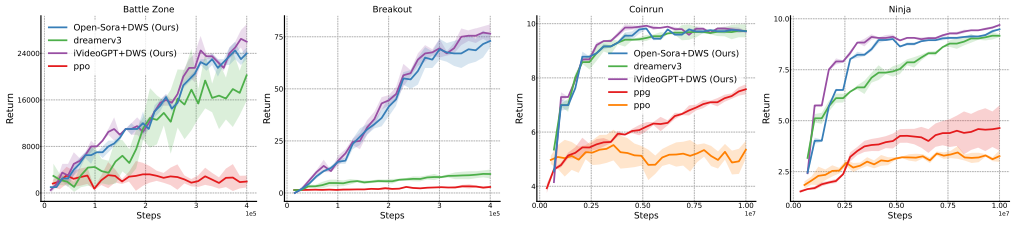
Qualitative Results Analysis. We qualitatively evaluate two different models, i.e., Open-Sora and iVideoGPT, fine-tuned by DWS using unseen initial frames. Fig. 4 showcases examples of video generations across diverse domains given unseen inputs. We observe that both base models fine-tuned by DWS successfully generate high-quality, controllable videos characterized by coherent temporal dynamics and consistent background preservation. Furthermore, the models demonstrate robust generalization capabilities when processing unseen inputs with varying backgrounds and textures, validating the effectiveness of our approach. As evidenced in Figure 5, while the base models tend to generate videos with visual distortions, DWS significantly enhances the output quality by maintaining object consistency and producing precise visual predictions.

Quantitative Results Analysis. From the results shown in Table 1 and Table 2, we have the following key observations: (i) DWS demonstrates superior performance in action-conditioned video prediction across different base models. On the BAIR dataset with a continuous action space, DWS significantly enhances the performance of both the diffusion-based Open-Sora and the autoregressive-based iVideoGPT, highlighting its generalizability and versatility across different architectures. The results show that Open-Sora, a traditional text-to-video model that conditions generation on static, global text prompts, can gain significant improvements in action controllability after DWS fine-tuning. Specifically, we observe an 11.7% reduction in FVD and a 27.9% decrease in LPIPS. Even with iVideoGPT, which is specifically designed for action-conditioned video generation, DWS achieves notable performance improvements. Regarding both Procgen and Atari game datasets with discrete action spaces, DWS consistently improves the base models' performance by a distinct margin, yielding significant enhancements in generated video quality. These enhancements can be attributed to two key components: The action-conditioned module, which efficiently modulates each reaction to its corresponding frame, and the motion-reinforced loss function, which effectively captures frame-to-frame pixel dynamics. (ii) DWS demonstrates its versatility as a universal method that can be efficiently deployed across diverse datasets, ranging from robotics datasets with continuous action spaces to game datasets with discrete action spaces. Furthermore, DWS can be seamlessly integrated into various architectures, where the two popular architectures considered in this work are diffusion-based and autoregressive transformer-based models.

5.2 Model-Based Reinforcement Learning

To validate the practical utility of DWS-fine-tuned models for world simulators, we evaluate their performance by utilizing them as world models in MBRL policy learning.

Benchmarks and Baselines. We conduct experiments on coinrun and ninja platformer games from the Procgen benchmark, and Breakout and Battle Zone games from the Atari benchmark. These environments take RBG images as observations and discrete actions for control. Procgen employs a 15-dimensional action space, Breakout operates with a 4-dimensional space, while Battle Zone utilizes an 18-dimensional action space. We compare our MBRL method with prioritized imagination with the following baselines: 1) PPO (Schulman et al., 2017) is a model-free RL method that is widely used. Our method is built on PPO. 2) Dreamerv3 (Hafner et al., 2023) is a SOTA model-based RL method that employs a recurrent network for dynamic prediction and actor-critic RL for policy learning, which is effective in handling tasks with discrete action spaces. For Procgen environments, we additionally include PPG (Cobbe et al., 2021) for comparison, as it represents a competitive algorithm on Procgen.



(a) Atari (b) Procgen

Figure 6: Averaged Return across five random seeds on Atari and Procgen environments.

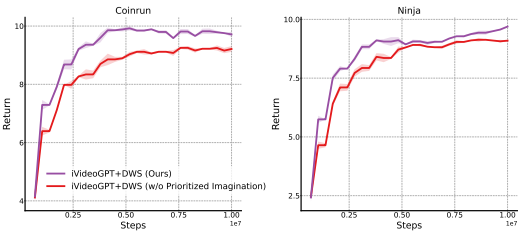


Figure 7: Prioritized imagination improves the performance of model-based RL.

Results Analysis. The experimental results presented in Fig. 6 demonstrate that our DWS-trained world model, when combined with a simple PPO algorithm, significantly outperforms both vanilla PPO and state-of-the-art model-based reinforcement learning methods, i.e., Dreamerv3. In Procgen environments, DWS exhibits substantial performance improvements over model-free approaches such as PPO and PPG. Although the DWS-trained world model requires fine-tuning during MBRL policy training—due to the episodic variations in background and object details inherent to Procgen, it maintains competitive performance compared to existing model-based methods. In Atari environments, DWS demonstrates substantial performance improvements over existing methods, attributed to its world models having acquired comprehensive dynamics knowledge for action simulation. Specifically, in Breakout, DWS achieves a remarkable $7\times$ performance gain compared to SOTA methods. This superior performance, particularly in terms of sample efficiency, can be attributed to our proposed prioritized imagination technique. We validate this contribution through ablation studies conducted on Procgen environments, with results presented in Figure 7.

5.3 Offline Model-Based Reinforcement Learning

We further explore the potential of leveraging trained video world models to augment offline datasets for policy enhancement. Using CoinRun and Ninja environments as case studies, we first establish baseline datasets by collecting 1M expert trajectories for each environment using a well-trained PPO agent, following Mediratta et al. (2024). For each evaluated environment, we employ *Open-Sora+DWS* to synthesize an additional 1M state-action transitions during training, effectively doubling the size of the original datasets. To validate the effectiveness of this data augmentation approach, we evaluate the performance improvements using two different offline RL algorithms: Conservative Q-Learning (CQL) (Kumar et al., 2020) and Implicit Q-Learning (IQL) (Kostrikov et al., 2022). As shown in Table 3, augmenting offline RL with world model-generated data during training significantly enhances performance across different algorithms and environments. These results demonstrate that the DWS-trained world simulator can generate meaningful state-action-reward transitions that effectively supplement the offline dataset for policy learning. The success of this approach highlights the potential of using pre-trained video models as world simulators for reinforcement learning applications. Furthermore, these results validate that DWS-trained models can generate accurate dynamic transitions, making them valuable tools for policy learning.

Table 3: Average return across 3 seeds on Coinrun and Ninja tasks.

Tasks	CQL	CQL w/ wm	IQL	IQL w/ wm
Coinrun	8.58 ± 0.29	$\mathbf{8.81 \pm 0.21}$(↑)	8.52 ± 0.26	$\mathbf{8.93 \pm 0.19}$(↑)
Ninja	5.92 ± 0.2	$\mathbf{6.31 \pm 0.23}$(↑)	5.7 ± 0.35	$\mathbf{6.33 \pm 0.17}$(↑)

6 Conclusion and Limitation

In this paper, we present DWS, a novel approach that efficiently adapts pre-trained video generative models as world simulators by leveraging their rich prior knowledge learned from large-scale datasets for downstream action-conditioned simulation. Our framework introduces a lightweight action-conditioned module that enables action-frame alignment and can be seamlessly integrated into various model architectures, and a motion-reinforced loss specifically designed to model inter-frame

pixel dynamics crucial for accurate world simulation. Extensive experimental results demonstrate that DWS significantly improves both video prediction quality and MBRL performance, leading to meaningful applications. However, DWS currently is limited in modeling videos with extended temporal horizons or high spatial resolutions. We leave it as future work.

REFERENCES

Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in atari. In *Thirty-eighth Conference on Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2405.12399.

Mohammad Babaeizadeh, Mohammad Taghi Saffar, Suraj Nair, Sergey Levine, Chelsea Finn, and Dumitru Erhan. Fitvid: Overfitting in pixel-level video prediction. *arXiv preprint arXiv:2106.13195*, 2021.

Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *IEEE International Conference on Computer Vision*, 2021.

Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47: 253–279, 2013.

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.

Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024.

Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, Yusuf Aytar, Sarah Maria Elisabeth Bechtle, Feryal Behbahani, Stephanie C.Y. Chan, Nicolas Heess, Lucy Gonzalez, Simon Osindero, Sherjil Ozair, Scott Reed, Jingwei Zhang, Konrad Zolna, Jeff Clune, Nando de Freitas, Satinder Singh, and Tim Rocktäschel. Genie: Generative interactive environments. In *Forty-first International Conference on Machine Learning*, 2024.

Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308, 2017.

Haoxuan Che, Xuanhua He, Quande Liu, Cheng Jin, and Hao Chen. Gamegen-$\mathbb{X}$: Interactive open-world game video generation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=8VG8tpPZhe.

Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis, 2023.

Tsai-Shien Chen, Aliaksandr Siarohin, Willi Menapace, Ekaterina Deyneka, Hsiang-wei Chao, Byung Eun Jeon, Yuwei Fang, Hsin-Ying Lee, Jian Ren, Ming-Hsuan Yang, and Sergey Tulyakov. Panda-70m: Captioning 70m videos with multiple cross-modality teachers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Karl Cobbe, Chris Hesse, Jacob Hilton, and John Schulman. Leveraging procedural generation to benchmark reinforcement learning. In *International conference on machine learning*, pp. 2048–2056. PMLR, 2020.

Karl W Cobbe, Jacob Hilton, Oleg Klimov, and John Schulman. Phasic policy gradient. In *International Conference on Machine Learning*, pp. 2020–2027. PMLR, 2021.

Decart, Julian Quevedo, Quinn McIntyre, Spruce Campbell, Xinlei Chen, and Robert Wachen. Oasis: A universe in a transformer. 2024. URL https://oasis-model.github.io/.

Zihan Ding, Amy Zhang, Yuandong Tian, and Qinqing Zheng. Diffusion world model. *arXiv preprint arXiv:2402.03570*, 2024.

Frederik Ebert, Chelsea Finn, Alex X Lee, and Sergey Levine. Self-supervised visual planning with temporal skip connections. *CoRL*, 12(16):23, 2017.

Ruili Feng, Han Zhang, Zhantao Yang, Jie Xiao, Zhilei Shu, Zhiheng Liu, Andy Zheng, Yukun Huang, Yu Liu, and Hongyang Zhang. The matrix: Infinite-horizon world generation with real-time moving control. *arXiv preprint arXiv:2412.03568*, 2024.

Agrim Gupta, Stephen Tian, Yunzhi Zhang, Jiajun Wu, Roberto Martín-Martín, and Li Fei-Fei. Maskvit: Masked visual pre-training for video prediction. *arXiv preprint arXiv:2206.11894*, 2022.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pp. 2555–2565. PMLR, 2019.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations*, 2020a. URL https://openreview.net/forum?id=S1lOTC4tDS.

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

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

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

Shengyi Huang, Rousslan Fernand Julien Dossa, Antonin Raffin, Anssi Kanervisto, and Weixun Wang. The 37 implementation details of proximal policy optimization. In *ICLR Blog Track*, 2022. URL https://iclr-blog-track.github.io/2022/03/25/ppo-implementation-details/. https://iclr-blog-track.github.io/2022/03/25/ppo-implementation-details/.

Quan Huynh-Thu and Mohammed Ghanbari. Scope of validity of psnr in image/video quality assessment. *Electronics Letters*, 44:800–801, 2008. URL https://api.semanticscholar.org/CorpusID:62732555.

Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. *Advances in neural information processing systems*, 32, 2019a.

Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. *Advances in neural information processing systems*, 32, 2019b.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=68n2s9ZJWF8.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. *Advances in Neural Information Processing Systems*, 33:1179–1191, 2020.

Bin Lin, Yunyang Ge, Xinhua Cheng, Zongjian Li, Bin Zhu, Shaodong Wang, Xianyi He, Yang Ye, Shenghai Yuan, Liuhan Chen, et al. Open-sora plan: Open-source large video generation model. *arXiv preprint arXiv:2412.00131*, 2024.

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*, 2022.

Xin Ma, Yaohui Wang, Gengyun Jia, Xinyuan Chen, Ziwei Liu, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. Latte: Latent diffusion transformer for video generation. *arXiv preprint arXiv:2401.03048*, 2024.

Marlos C Machado, Marc G Bellemare, Erik Talvitie, Joel Veness, Matthew Hausknecht, and Michael Bowling. Revisiting the arcade learning environment: Evaluation protocols and open problems for general agents. *Journal of Artificial Intelligence Research*, 61:523–562, 2018.

Ishita Mediratta, Qingfei You, Minqi Jiang, and Roberta Raileanu. The generalization gap in offline reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=3w6xuXDOdY.

Vincent Micheli, Eloi Alonso, and François Fleuret. Transformers are sample-efficient world models. In *The Eleventh International Conference on Learning Representations*, 2023a. URL https://openreview.net/forum?id=vhFu1Acb0xb.

Vincent Micheli, Eloi Alonso, and François Fleuret. Transformers are sample-efficient world models. In *The Eleventh International Conference on Learning Representations*, 2023b. URL https://openreview.net/forum?id=vhFu1Acb0xb.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

Jack Parker-Holder, Philip Ball, Jake Bruce, Vibhavari Dasagi, Kristian Holsheimer, Christos Kaplanis, Alexandre Moufarek, Guy Scully, Jeremy Shar, Jimmy Shi, Stephen Spencer, Jessica Yung, Michael Dennis, Sultan Kenjeyev, Shangbang Long, Vlad Mnih, Harris Chan, Maxime Gazeau, Bonnie Li, Fabio Pardo, Luyu Wang, Lei Zhang, Frederic Besse, Tim Harley, Anna Mitenkova, Jane Wang, Jeff Clune, Demis Hassabis, Raia Hadsell, Adrian Bolton, Satinder Singh, and Tim Rocktäschel. Genie 2: A large-scale foundation world model. 2024.

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.

Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024.

Marc Rigter, Tarun Gupta, Agrin Hilmkil, and Chao Ma. Avid: Adapting video diffusion models to world models. *arXiv preprint arXiv:2410.12822*, 2024.

Jan Robine, Marc Höftmann, Tobias Uelwer, and Stefan Harmeling. Transformer-based world models are happy with 100k interactions. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=TdBaDGCpjly.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Abhishek Sharma, Adams Yu, Ali Razavi, Andeep Toor, Andrew Pierson, Ankush Gupta, Austin Waters, Aäron van den Oord, Daniel Tanis, Dumitru Erhan, Eric Lau, Eleni Shaw, Gabe Barth-Maron, Greg Shaw, Han Zhang, Henna Nandwani, Hernan Moraldo, Hyunjik Kim, Irina Blok, Jakob Bauer, Jeff Donahue, Junyoung Chung, Kory Mathewson, Kurtis David, Lasse Espeholt, Marc van Zee, Matt McGill, Medhini Narasimhan, Miaosen Wang, Mikołaj Bińkowski, Mohammad Babaeizadeh, Mohammad Taghi Saffar, Nando de Freitas, Nick Pezzotti, Pieter-Jan Kindermans, Poorva Rane, Rachel Hornung, Robert Riachi, Ruben Villegas, Rui Qian, Sander Dieleman, Serena Zhang, Serkan Cabi, Shixin Luo, Shlomi Fruchter, Signe Nørly, Srivatsan Srinivasan, Tobias Pfaff, Tom Hume, Vikas Verma, Weizhe Hua, William Zhu, Xinchen Yan, Xinyu Wang, Yelin Kim, Yuqing Du, and Yutian Chen. Veo. 2024. URL https://deepmind.google/technologies/veo/.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Richard S Sutton. Dyna, an integrated architecture for learning, planning, and reacting. *ACM Sigart Bulletin*, 2(4):160–163, 1991.

Richard S. Sutton and Andrew G. Barto. Reinforcement learning - an introduction. In *Adaptive computation and machine learning*, 1998. URL https://api.semanticscholar.org/CorpusID:264703640.

Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, et al. Deepmind control suite. *arXiv preprint arXiv:1801.00690*, 2018.

Keyu Tian, Yi Jiang, Zehuan Yuan, BINGYUE PENG, and Liwei Wang. Visual autoregressive modeling: Scalable image generation via next-scale prediction. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=gojL67CfS8.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 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.

Dani Valevski, Yaniv Leviathan, Moab Arar, and Shlomi Fruchter. Diffusion models are real-time game engines. *arXiv preprint arXiv:2408.14837*, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

Vikram Voleti, Alexia Jolicoeur-Martineau, and Chris Pal. Mcvd-masked conditional video diffusion for prediction, generation, and interpolation. *Advances in neural information processing systems*, 35:23371–23385, 2022.

Jialong Wu, Shaofeng Yin, Ningya Feng, Xu He, Dong Li, Jianye Hao, and Mingsheng Long. ivideogpt: Interactive videogpts are scalable world models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Jiannan Xiang, Guangyi Liu, Yi Gu, Qiyue Gao, Yuting Ning, Yuheng Zha, Zeyu Feng, Tianhua Tao, Shibo Hao, Yemin Shi, et al. Pandora: Towards general world model with natural language actions and video states. *arXiv preprint arXiv:2406.09455*, 2024.

Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. Videogpt: Video generation using vq-vae and transformers. *arXiv preprint arXiv:2104.10157*, 2021.

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

Mingyu Yang, Junyou Li, Zhongbin Fang, Sheng Chen, Yangbin Yu, Qiang Fu, Wei Yang, and Deheng Ye. Playable game generation. *arXiv preprint arXiv:2412.00887*, 2024a.

Sherry Yang, Jacob C Walker, Jack Parker-Holder, Yilun Du, Jake Bruce, Andre Barreto, Pieter Abbeel, and Dale Schuurmans. Position: Video as the new language for real-world decision making. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 56465–56484. PMLR, 21–27 Jul 2024b.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024c.

Jiwen Yu, Yiran Qin, Xintao Wang, Pengfei Wan, Di Zhang, and Xihui Liu. Gamefactory: Creating new games with generative interactive videos. *arXiv preprint arXiv:2501.08325*, 2025.

Lijun Yu, Yong Cheng, Kihyuk Sohn, José Lezama, Han Zhang, Huiwen Chang, Alexander G Hauptmann, Ming-Hsuan Yang, Yuan Hao, Irfan Essa, et al. Magvit: Masked generative video transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10459–10469, 2023.

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.

Weipu Zhang, Gang Wang, Jian Sun, Yetian Yuan, and Gao Huang. STORM: Efficient stochastic transformer based world models for reinforcement learning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=WxnrX42rnS.

Yang Zhang, Chenjia Bai, Bin Zhao, Junchi Yan, Xiu Li, and Xuelong Li. Decentralized transformers with centralized aggregation are sample-efficient multi-agent world models. *arXiv preprint arXiv:2406.15836*, 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, March 2024. URL https://github.com/hpcaitech/Open-Sora.

Fangqi Zhu, Hongtao Wu, Song Guo, Yuxiao Liu, Chilam Cheang, and Tao Kong. Irasim: Learning interactive real-robot action simulators. *arXiv:2406.12802*, 2024.

A Details of Dataset Collection

A.1 Atari

In our experiments on two Atari games - *Breakout* and *Battle Zone*, we utilized a well-established baseline for Atari, Deep Q-Learning (DQN) agent (Mnih et al., 2013), to collect the offline dataset for tuning. Specifically, we adopted an open-source DQN implementation provided at `https://github.com/vwxyzjn/cleanrl`, trained the agent for 1 million environment steps, and stored the replay buffer as the offline dataset. Since our work focuses on developing a more accurate world model, we intentionally initialized these two game environments without enabling random sticky actions. This ensures that the collected trajectories accurately capture the ground-truth state transitions of the environment. Further, we processed the initialized environments by applying a FrameSkipping wrapper with a frame skip of 4 and a ResizingObservation wrapper, which resizes the output RGB observations to a resolution of $(84, 84)$ for DQN agent training. In terms of the hyperparameter of DQN, we followed the default setting provided at `https://github.com/vwxyzjn/cleanrl/blob/master/cleanrl/dqn_atari.py`.

A.2 Procgen

Following a similar approach to our Atari collection experiments, we collected replay datasets from two Procgen games: Coinrun and Ninja. The data collection process utilized a publicly available Proximal Policy Optimization (PPO) implementation (Huang et al., 2022), accessed through `https://github.com/vwxyzjn/cleanrl/blob/master/cleanrl/ppo_procgen.py`. For each game, we collected 1 million state transitions at the default Procgen resolution of 64×64 pixels for DWS training.

B Implementation Details

B.1 Details of Base Models

Open-Sora. Open-Sora, based on rectified-flow (Liu et al., 2022) and spatial-temporal transformer (Ma et al., 2024; Chen et al., 2023) architecture , is a text-and-frame-conditioned video generation model. For our implementation of DWS, we utilized Open-Sora version 1.2 as our base model. To maintain consistency with the original architecture, we employed the same Variational Autoencoder (VAE) provided in the Open-Sora library. However, to optimize computational efficiency, we adopted T5-small for text encoding instead of the original text encoder. Given that the original Open-Sora model comprises 1.1 billion parameters, which poses significant computational constraints for inference and downstream applications, we strategically initialized our model using only the first 12 layers of the Open-Sora architecture. This modification substantially reduced the model's computational footprint while preserving essential generative capabilities. Our proposed action-conditioned module is integrated into each layer of Open-Sora to improve action-frame alignment.

iVideoGPT. iVideoGPT is an autoregressive transformer-based architecture that builds upon the LLaMA architecture (Touvron et al., 2023). The model extends the base architecture by incorporating specialized reward prediction head-layers, enabling it to perform both video generation and reward estimation tasks. A distinctive feature of iVideoGPT is its context-aware tokenizer, which implements compressive tokenization for efficient video prediction. We initialize our model with weights from `https://huggingface.co/thuml/ivideogpt-oxe-64-act-free`, including a tokenizer of 114M size and a transformer of 138M size. Our proposed action-conditioned module is integrated into each transformer block.

B.2 Implementation of Motion-Reinforced Loss

For diffusion-based models, which use a squared l_2 distance for computing loss function and conducting gradient updates, their original loss functions can be represented as follows:

$$\mathcal{L}_{\text{prev}} = \mathbb{E}[\|g - p_\theta(x_t, t, y)\|_2^2], \tag{4}$$

where y is the condition, and g represents the target signal to be estimated, which takes different forms depending on the model architecture: For DDPM-style diffusion models (Ho et al., 2020),

it corresponds to the ground-truth noise, while for flow-matching-based models (Liu et al., 2022), it represents the velocity. Given that the output of p_θ and the video embedding x_t share identical dimensionality, thus we can incorporate the motion-based weights ω into $\mathcal{L}_{\text{prev}}$ as follows:

$$\mathcal{L}_{\text{motion}} = \mathbb{E}[\omega \| g - p_\theta(x_t, t, y) \|_2^2]. \tag{5}$$

For autoregressive transformer-based models, which use a cross-entropy loss for training, their original loss functions can be represented as follows:

$$\mathcal{L}_{\text{prev}} = -\sum_{i=t}^{L} \log p(x_i | x_{<i}), \tag{6}$$

where x_i denotes discrete tokens embedded in the transformer, and L is the sequence length. In this case, we compute ω as follows:

$$\omega_{i+1} = c^{\mathbb{I}(x_{i+1}=x_i)}, \tag{7}$$

where $c = e$ and $\mathbb{I}$ denotes an indicator function. Thus, we obtain the final $\mathcal{L}_{\text{motion}}$ for transformer-based models:

$$\mathcal{L}_{\text{motion}} = -\sum_{i=t}^{L} \omega \log p(x_i | x_{<i}), \tag{8}$$

B.3 Implementation of the MBRL Algorithm

We develop a simple model-based RL algorithm using a DWS-trained world model within the MBPO framework (Janner et al., 2019a; Wu et al., 2024), with PPO (Schulman et al., 2017) as the base actor-critic RL algorithm. We provide the pseudo-code in Alg. 1. We build our codes upon the implementation in `https://github.com/vwxyzjn/cleanrl/blob/master/cleanrl/ppo_procgen.py` for Procgen environments and `https://github.com/vwxyzjn/cleanrl/blob/master/cleanrl/ppo_atari.py` for Atari environments, using the same hyperparameters and architecture for actor-critic learning. Hyperparameters specific to model-based RL are listed in Table 4. Following Dreamerv3 (Hafner et al., 2023), we use a symlog transformation for reward prediction.

Algorithm 1 Model-Based Reinforcement Learning (Janner et al., 2019a) with Prioritized Imagination

Initialize real replay buffer $\mathcal{B}_{\text{real}}$ with random policy
Initialize actor-critic π_ϕ, v_ϕ, world model p_θ
Initially train model p_θ on $\mathcal{B}_{\text{real}}$
Initialize imagined replay buffer $\mathcal{B}_{\text{imag}}$
for N steps **do**
  // Training
  if world model update **then**
    Update world model p_θ on a mini-batch from $\mathcal{B}_{\text{real}}$
  end if
  Compute TD-error δ, loss $\mathcal{L}_{\text{rl}}$ with PPO on a mini-batch from $\mathcal{B}_{\text{imag}} \cup \mathcal{B}_{\text{real}}$
  Update actor-critic π_ϕ, v_ψ
  Update priorities of samples in mini-batch from $\mathcal{B}_{\text{real}}$ with the new values as $max(\delta, 0) + \epsilon$ // $\epsilon = 1e^{-6}$
  // Data collection
  if world model rollout **then**
    Sample a mini-batch of o_t from $\mathcal{B}_{\text{real}}$ based on priorities
    Perform k-step model rollout starting from o_t using policy π_ϕ; add to $\mathcal{B}_{\text{imag}}$
  end if
  Take action in environment according to π_ϕ; add to $\mathcal{B}_{\text{real}}$
end for

Hyperparameter	Value
World model rollout batch size	16
World model rollout horizon	9
Real data ratio	0.5
World model training batch size	16
World model training learning rate	2e-5
Optimizer	Adam

Table 4: Hyperparameters of our proposed model-based RL algorithm with prioritized imagination.