Self-Improving World Modelling with Latent Actions

Anonymous Author(s)
Affiliation
Address
email

Abstract

Learning to model the world—predicting transitions between previous states X and next states Y under actions Z—is essential to reasoning and planning in AI; however, it typically requires costly action-labelled state trajectories. We propose SWIRL, a self-improvement framework for LLM and VLM post-training that learns from state-only sequences by treating actions as a latent variable. SWIRL alternates between Forward World Modelling (FWM) $P_\theta(Y|X, Z)$ and an Inverse Dynamics Modelling (IDM) $Q_\phi(Z|X, Y)$. These two phases correspond to: (1) Variational Information Maximisation, which updates the FWM to generate next states that maximise conditional mutual information with latent actions given prior states, encouraging identifiable consistency; and (2) ELBO Maximisation, which updates the IDM to explain observed transitions, effectively performing coordinate ascent. Both models are trained with reinforcement learning (specifically, GRPO) with the opposite frozen model's log-probability as a reward signal. We provide theoretical learnability guarantees for both updates, evaluating SWIRL on LLMs and VLMs across multiple environments: single-turn and multi-turn open-world visual dynamics and synthetic textual environments for physics, web, and tool calling. SWIRL achieves gains of 15% on AURORA-BENCH, 28% on ByteMorph, 3% on WORLDPREDICTIONBENCH, and 14% on STABLETOOLBENCH.[1]

1 Introduction

World modelling is a model's understanding of the environment and agent dynamics, i.e., trajectories of states and actions, which enables simulating possible futures without hallucinations. Foundation models, including Large Language Models (LLMs) and Vision–Language Models (VLMs), have arguably been shown to internalise world modelling to some extent during their training, thus enabling reasoning and planning without an external, specialised world model [1–6]. For example, explicitly training models to predict the results of invoking tools [7] or executing code [8] significantly enhances applications that require these skills. This ability can also transfer to spatial reasoning tasks [9, 10] and reduce hallucinations that contradict external regularities [11, 4].

While promising, the development of robust internal world models faces a significant bottleneck in data scalability. Current approaches rely heavily on execution logs or trajectories where observations are densely annotated with specific actions, as such annotations are naturally available in restricted environments (e.g., logs for tool calling or coding). However, for open-world tasks, collecting manual annotations for every transition is prohibitively expensive and intractable. Moreover, an additional challenge is the ambiguity of *inverse dynamics*: a transition between two states may be explained by multiple valid actions, making purely supervised learning brittle when data is sparse.

Inspired by recent advances in self-improving learning for other applications [12–15, 14], as well as world models with discrete latent actions trained from scratch [16, 17], we propose SWIRL

[1]The code and models developed in this paper will be made available at <Anonymous>.

Submitted to 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Do not distribute.

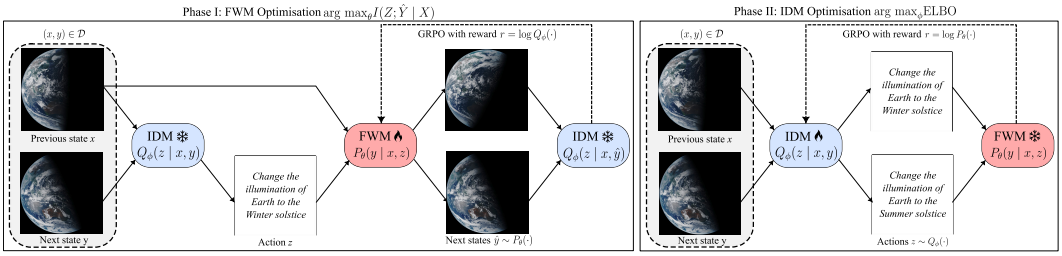


Figure 1: In SWIRL, we facilitate the world modelling ability of foundation models (LLMs and VLMs) by modelling two components: Forward World Model (FWM) $P_\theta(y \mid x, z)$ and Inverse Dynamics Model (IDM) $Q_\phi(z \mid x, y)$. These components are iteratively optimised through GRPO in two distinct phases: I) the FDM acts as a policy and the IDM as a reward to ensure identifiability between actions and next states; II) the IDM acts as a policy and the FDM as a reward to ensure data fidelity to the state-only sequences. The KL term is omitted from the figure for simplicity.

(Self-improving World modelling with Iterative RL), an iterative post-training framework to enhance the intrinsic world modelling of LLMs and VLMs from state-only sequences. Specifically, we take advantage of the inductive bias offered by foundation model pre-training to treat actions as latents grounded in natural language. In addition, we enhance *Forward World Modelling* (FWM; predicting the next state y given current state and latent action x, z) via *Inverse Dynamics Model* (IDM; inferring latent action z given states x, y), and vice versa [18, 9, 19]. Specifically, in our reciprocal framework, we optimise the FWM P_θ to generate futures that are consistently identifiable by the IDM, and the IDM Q_ϕ to infer actions that maximise the likelihood of the state dynamics predicted by the FWM.

From a variational inference perspective, we theoretically prove that the optimisation of FWM is equivalent to maximising a lower bound on the Conditional Mutual Information $I(Z; \hat{Y}|X)$ [20] between the latent Z and future states $\hat{Y}$ predicted by the FWM; and that the optimisation of the IDM is equivalent to performing coordinate ascent on the Evidence Lower Bound (ELBO) of the log-likelihood $\log P_\theta(Y|X)$. Without relying on ground-truth action annotations, we optimise such a reciprocal framework with Group Relative Policy Optimisation (GRPO; Guo et al. 21). FWM and IDM take alternating roles of policy and reward, iterating until both components converge.

We validate our framework across four distinct environments on six benchmarks: open-world visual dynamics for VLMs on AURORA-BENCH, BYTEMORPH and WORLDPREDICTIONBENCH, synthetic textual worlds on SCIENCEWORLD, web HTML on MIND2WEB, and tool calling on STABLETOOL-BENCH for LLMs. Empirical results confirm that our reciprocal self-improving post-training allows models to learn effective dynamics from unlabelled state sequences, outperforming supervised fine-tuning baselines and achieving parity with larger state-of-the-art models. Our contributions are summarised as follows:

- We propose a novel self-improving framework for world modelling in LLMs and VLMs, reciprocally reinforcing FWM and IDM without action annotations.
- We derive a rigorous theoretical proof that SWIRL corresponds to alternating between maximising Variational Mutual Information and Evidence Lower Bound.
- We provide empirical evidence on six benchmarks across visual, textual, web, and tool calling environments, demonstrating the effectiveness of SWIRL to enhance world modelling capabilities.

2 Related Work

Intrinsic World Models. Recent research has explored through careful evaluation whether internalised world models emerge in LLMs and VLMs. Vafa et al. [1] assessed whether the model representations truly capture coherent world dynamics [6]. Similarly, Tehenan et al. [10] and Qiu et al. [2] found evidence that LLMs implicitly encode spatial and temporal relationships to some degree. World modelling also emerges naturally during the pre-training of large unified VLMs [22, 23] and from video-based training [4, 9].

In addition, previous work established that explicit world modelling can boost performance in downstream applications. For instance, Copet et al. [8] proposed the Coding World Model for programming, and Lehrach et al. [24] extended this approach to game-playing environments. Modelling

the outcomes of function tool calls [7] or planning [25] had similar effects. As a result, dedicated post-training pipelines have been proposed to endow LLMs [26] and VLMs [27] with explicit world modelling capabilities. Extensive benchmarks for evaluation have been proposed, including forward and inverse dynamics prediction [19, 18, 28].

Latent Action World Models. A parallel line of work trains world models directly from observation-only sequences by discovering latent actions without annotations. This was pioneered by Genie [16], which employed a two-stage pipeline where a smaller IDM is used to discover actions before freezing them for the larger FWM. Subsequent methods improved cross-environment adaptability [29], replacing the sequential pipeline with a joint co-evolution of these two models [17], and unified comprehension, video synthesis, and action prediction in a single architecture [30]. More recently, the paradigm has been extended to in-the-wild videos [31], cross-embodiment robotic manipulation via diffusion policies [32], and distractor-robust training via segmentation guidance [33]. While these methods share our motivation of dispensing with action labels, they operate on pixel-level generative video models with reconstruction-based objectives; SWIRL instead grounds latent actions in *natural language* and applies a *reciprocal* reinforcement learning loop to jointly improve FWM and IDM based on the other model's rewards within LLMs and VLMs.

Self-Improving Learning. The paradigm of self-improving learning refers to models learning from their own (possibly curated) signals without external supervision. Huang et al. [34] showed that LLMs can generate high-confidence answers and fine-tune themselves on these outputs to improve reasoning. Bensal et al. [35] proposed a self-reflection and reinforcement learning loop where models analyse mistakes and retry, boosting performance on tasks like maths and function calling. Lee et al. [36] introduce a curriculum in which models iteratively generate and filter correct answers, progressively tackling harder problems, while Zhao et al. [37] demonstrate that self-synthesised input–output pairs improve classification and generation quality. Wang et al. [12] enhance self-improvement capabilities of agents with a skill library. LLM's coding and unit test generation capabilities can also co-evolve iterating on each other's outcomes [13].

In unified VLMs [22, 38–40], which can interleave text and images, understanding performance often exceeds generation [41–45]. A common strategy is to use the understanding head as a critic to guide generation with carefully designed rubrics or heuristics [15, 14, 46]. Unlike these approaches, SWIRL theoretically and empirically proves the effectiveness of utilising the predicted likelihoods from understanding (action prediction) and generation (next state prediction) to establish a reciprocal, virtuous cycle for VLM improvement.

3 Methodology

3.1 Task Formulation

We consider transitions from a source state $x \in \mathcal{S}$ to a target state $y \in \mathcal{S}$, mediated by a latent action $z \in \mathcal{A}$. Following Qiu et al. [9] and Wang et al. [18], we parametrise world modelling using two components: i) *Forward World Modelling (FWM)*: $P_\theta(y|x, z)$, which predicts the next state given previous state and action; and ii) *Inverse-Dynamics Prediction (IDM):* $Q_\phi(z|x, y)$, which infers the action given the state transition. The parameters θ and ϕ can be either disjoint or shared (as discussed in Appendix F).

We consider four classes of environments with different observation–action formulations. In real-world visual environments, observations are pixel-level visual inputs and actions are specified in natural language. We study this setting using unified vision–language models (VLMs) capable of perceiving and generating interleaved image–text sequences. In synthetic textual environments, both observations and actions are expressed purely in language and are governed by an underlying simulator, which we model using large language models (LLMs). In web-based environments, states and actions correspond to HTML and interaction logs, respectively, while in tool-use settings, actions are tool calls and observations consist of the conversational context and tool execution outcomes.

3.2 SWIRL

Intuition. SWIRL alternates between two tightly coupled optimisation phases. In the first phase, we optimise the Forward World Model (FWM), P_θ, to generate future states that are identifiable

Algorithm 1 SWIRL

Input: Unlabelled dataset $\mathcal{D} = \{(x_i, y_i)\}$ **Initialise:** FWM P_θ, IDM Q_ϕ **Hyperparams.:** Group size G, Learning rates η_θ, η_ϕ
repeat until convergence or max iterations:
=== Phase I: optimise FWM ===
Freeze IDM parameters ϕ.
for each batch $x \sim \mathcal{D}$ **do**
  Sample $z \sim Q_\phi(z|x, y)$.
  Rollouts: $\{\hat{y}_1, \ldots, \hat{y}_G\} \sim P_\theta(\cdot|x, z)$.
  for $k = 1$ to G **do**
    $R_k \leftarrow \log Q_\phi(z|x, \hat{y}_k)$
  Compute advantage A_k^{F} on FWM rollouts.
  $\theta \leftarrow \theta + \eta_\theta \nabla_\theta \left[\frac{1}{G}\sum_{k=1}^{G} A_k^{\mathrm{F}} \log P_\theta(\hat{y}_k|x, z)\right]$
=== Phase II: optimise IDM ===
Freeze FWM parameters θ.
for each batch $(x, y) \sim \mathcal{D}$ **do**
  Sample $\{z_1, \ldots, z_G\} \sim Q_\phi(\cdot|x, y)$.
  for $k = 1$ to G **do**
    $R_k \leftarrow \log P_\theta(y|x, z_k)$
  Compute advantage A_k^{I} on IDM rollouts.
  $\phi \leftarrow \phi + \eta_\phi \nabla_\phi \left[\frac{1}{G}\sum_{k=1}^{G} A_k^{\mathrm{I}} \log Q_\phi(z_k|x, y)\right]$ †
end repeat
Return optimised FWM θ^* and IDM ϕ^*

†The KL term is omitted in the GRPO objectives for brevity's sake.

by the Inverse Dynamics Model (IDM). This enforces **identifiability** in forward prediction. In Eq. 2 below, we prove that this objective is equivalent to maximising a variational lower bound on the conditional mutual information $I(Z; \hat{Y}|X)$ [20] , encouraging the predicted futures $\hat{Y}$ to retain maximal information about the underlying latent actions Z.

In the second phase, we optimise the IDM, Q_ϕ, to improve **data fidelity** by inferring actions that best explain the observed state transitions under the learned FWM. We formally justify this step by proving that the resulting objective corresponds to maximising the Evidence Lower Bound (ELBO) of the inverse model (see Eq. 4 below).

Objectives. As depicted in Algorithm 1, we rely only on state-only sequences (x_t, y_{t+1}) as data. In the first phase of each iteration of SWIRL, we optimise FWM as a policy based on the frozen IDM rewards. Let P_θ denote the FWM policy parametrised by θ. For each training step in GRPO [21], we first sample a latent action $z_t \sim Q_\phi(\cdot \mid x_t, y_t)$ from the IDM. Then, we sample a group of G rollouts $\{\hat{y}_{t+1}^{(k)}\}_{k=1}^{G}$ from $P_\theta(\cdot \mid x_t, z_t)$. To evaluate their quality, we use a frozen IDM model Q_ϕ prompted for the IDM task, which estimates the likelihood of the action z_t given the generated transition: $P_{\pi_\phi}(z_t \mid x_t, \hat{y}_{t+1}^{(k)})$. The reward for the k-th rollout is defined as

$$r_k = \log Q_\phi(z_t \mid x_t, \hat{y}_{t+1}^{(k)}).$$

This reward enforces that a predicted future is considered plausible only if IDM can retrospectively explain the action that led to the predicted state. We then optimise P_θ to maximise the advantage induced by these rewards.

After the FWM policy converges, we invert the optimisation direction to refine the IDM. In this phase, the optimised FWM model is frozen and used as the reward signal. The policy Q_ϕ is prompted for the IDM task, generating action candidates $\{\hat{z}_t^{(i)}\}_{i=1}^{G}$ given a state transition (x_t, y_{t+1}). Each candidate action is evaluated by querying the FWM for the likelihood of the target observation:

$$r_k = \log P_{\pi_\theta}(y_{t+1} \mid x_t, \hat{z}_t^{(k)}).$$

This reward favours data fidelity with respect to the state transition. We alternate between FWM and IDM optimisation phases, swapping the roles of policy and reward model until convergence. This reciprocal training paradigm ensures that improvements in forward prediction directly enhance action inference by the inverse dynamic model, and vice versa, driving the system toward a globally consistent both given all unlabelled observations.

3.3 Theoretical Analysis

Phase I encourages that generated futures are *distinguishable* (high mutual information), allowing the FWM to produce distinct outcomes for varied latent actions. Phase II encourages that inferred actions are *plausible* (high likelihood), allowing the IDM to map transitions to actions that the FWM can reproduce.

Phase I: FWM optimisation via Variational Information Maximisation. In this phase, we freeze ϕ and update θ to generate next states $\hat{y}$ that are identifiable by the inference model. We formalise this as maximising the Conditional Mutual Information (CMI) $I(Z;\hat{Y}|X)$ with respect to the *empirical belief distribution* of the model. Let $\tilde{P}(z|x) \triangleq \mathbb{E}_{y\sim\mathcal{D}(y|x)}[Q_\phi(z|x,y)]$ denote the marginal distribution of latent actions inferred by the IDM over the dataset.

Theorem 3.1 (FWM Lower Bound)**.** *Optimising the FWM to maximise the log-probability assigned to generated samples by the frozen IDM maximises a variational lower bound on the Conditional Mutual Information $I_{\tilde{P}}(Z;\hat{Y}|X)$ defined over the empirical belief distribution $\tilde{P}(z|x)$.*

Proof. The mutual information under the data-induced joint distribution $P(x)\tilde{P}(z|x)P_\theta(\hat{y}|x,z)$ is:

$$I(Z;\hat{Y}|X) = \mathbb{E}_{x\sim\mathcal{D}}\left[H_{\tilde{P}}(Z|X) - H(Z|\hat{Y},X)\right]. \tag{1}$$

The entropy of the latent belief $H_{\tilde{P}}(Z|X)$ depends only on the frozen IDM and data distribution, and is thus constant w.r.t. θ. Maximising CMI is therefore equivalent to minimizing the conditional entropy $H(Z|\hat{Y},X)$. We use the variational posterior $Q_\phi(z|x,\hat{y})$ as a proxy for the intractable true posterior $P_\theta(z|\hat{y},x)$ [20] to bound the entropy term:

$$-H(Z|\hat{Y},X) = \mathbb{E}_{x,z\sim\tilde{P},\hat{y}\sim P_\theta}[\log P_\theta(z|\hat{y},x)] \geq \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{z\sim\tilde{P}(z|x)}\mathbb{E}_{\hat{y}\sim P_\theta(\cdot|x,z)}[\log Q_\phi(z|x,\hat{y})].$$

Substituting the definition of $\tilde{P}(z|x)$ creates the following objective:

$$\mathcal{J}(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\mathbb{E}_{z\sim Q_\phi(\cdot|x,y)}\mathbb{E}_{\hat{y}\sim P_\theta(\cdot|x,z)}[\log Q_\phi(z|x,\hat{y})]. \tag{2}$$

This matches Algorithm 1: we sample trajectories (x,y), infer z using the IDM, rollout $\hat{y}$ using the FWM, and reward the FWM with the IDM's log likelihood. The gradient of Eq. 2 is $\nabla_\theta\mathcal{J}(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\mathbb{E}_{z\sim Q_\phi(\cdot|x,y)}\mathbb{E}_{\hat{y}\sim P_\theta(\cdot|x,z)}[\log Q_\phi(z \mid x,\hat{y})\nabla_\theta \log P_\theta(\hat{y} \mid x,z)]$, for which Group Relative Policy Optimisation (GRPO) offers an estimator:

$$\widehat{\nabla_\theta\mathcal{J}}_{\text{GRPO}}(\theta) = \frac{1}{G}\sum_{k=1}^{G} A_k^{\text{F}}\nabla_\theta \log P_\theta(\hat{y}_k \mid x,z), \tag{3}$$

where A_k are the group-relative advantages derived from rewards $R_k = \log Q_\phi(z \mid x,\hat{y}_k)$. □

Phase II: IDM optimisation via ELBO Maximisation. In the second phase, we freeze θ and optimise ϕ. The goal is to infer actions z that explain the transition (x,y) in the FWM dynamics, while staying close to the valid prior. We treat the initialised model at each iteration as an *informative prior* $P(z|x) \triangleq \pi_{\text{ref}}(z|x)$. This corresponds to maximising the Evidence Lower Bound (ELBO).

Theorem 3.2 (IDM Lower Bound)**.** *Optimising the IDM via the KL-regularised policy gradient objective, with reward $R = \log P_\theta(y|x,z)$, $\beta = 1$, reference policy π_{ref} maximises the ELBO objective where the prior is defined by π_{ref}.*

Proof. We seek to maximise the marginal log-likelihood $\log P_\theta(y|x)$. By introducing the variational distribution $Q_\phi(z|x,y)$ and the informative prior $P(z|x) = \pi_{\text{ref}}(z|x)$, we derive the ELBO:

$$\begin{aligned}\log P_\theta(y|x) &= \log \mathbb{E}_{z\sim Q_\phi}\left[\frac{P_\theta(y|x,z)P(z|x)}{Q_\phi(z|x,y)}\right] \\ &\geq \mathbb{E}_{z\sim Q_\phi(\cdot|x,y)}\left[\log P_\theta(y|x,z)\right] - D_{\text{KL}}(Q_\phi(z|x,y)\,||\,P(z|x)) \triangleq \mathcal{L}_{\text{ELBO}}.\end{aligned}$$

The KL-regularised policy gradient objective is defined as the expected reward subject to a KL constraint weighted with $\beta = 1$ against the reference policy:[2]

$$\mathcal{J}(\phi) = \mathbb{E}_{z\sim Q_\phi}[R(x,z,y)] - D_{\text{KL}}(Q_\phi(\cdot|x,y)||\pi_{\text{ref}}(\cdot|x)). \tag{4}$$

By setting the reward $R(x,z,y) = \log P_\theta(y|x,z)$ and identifying the prior $P(z|x)$ with the reference policy $\pi_{\text{ref}}(z|x)$, we observe that $\mathcal{J}(\phi) \equiv \mathcal{L}_{\text{ELBO}}$. We again use the GRPO as the estimator to perform coordinate ascent on the ELBO. □

[2] In practice, we allow β to be tunable to adjust the deviation against the reference policy, as is common practice in GRPO.

4 Experiments and Results

4.1 Experimental Setup

Models. We choose Liquid [38] as our base VLM as the publicly best 7B-size unified VLM with an autoregressive architecture. The medium size limits the scale requirements for the compute infrastructure, and the autoregressive nature allows for applying GRPO off-the-shelf (without *ad-hoc* adaptations for diffusion-based generation). For text-based environments, we use Qwen-2.5-3B-Instruct [47], a competitive mid-size LLM.

SFT Warm-up. Since our base VLMs/LLMs are general-purpose, they lack the specific interface capabilities required for environments (e.g., Liquid cannot generate images using a combination of image and text inputs). Prior to SWIRL, we conduct an initial SFT, which is restricted as policy initialisation to ensure the model outputs valid modalities; without this, a random policy would render RL exploration impossible (as Liquid's poor zero-shot performance in GEDIT-BENCH in Appendix E). However, a stronger backbone that natively supports image-and-text-conditioned image generation (e.g., BAGEL or Emu3.5) could be plugged into SWIRL, bypassing this stage entirely. For VLMs, we utilise image editing mixtures from PICO-BANANA-400K [48] and AURORA [49]. For LLMs, we fine-tune on half of the environment-specific episodes, and leave the rest for SWIRL.

Iterative RL. We first conduct controlled experiments on the unlabelled video mixture from UCF-101 [50], Movement-in-Times [51], Kinetics700 [52], limiting training to the first phase, i.e., SWIRL (IDM→FWM). This setup enables strict comparison with the bootstrapping baseline of Qiu et al. [9] and allows us to observe stable convergence within a single epoch. For the iterative SWIRL, we uniformly sample 30K videos per iteration from a large-scale unlabelled video corpus, VIDGEN-1M [53], and extract the frame pairs as in [54]. Each iteration is trained for one epoch, and the model alternates between FWM and IDM optimisation phases as described in Section 3.2. θ and ϕ (FWM and IDM parameters) are shared in our experiments.

Benchmarks. We evaluate visual world modelling under two settings. Firstly, for single-turn next-observation prediction, we use AURORA-BENCH and BYTEMORPH, both of which formulate dynamics prediction as action-conditioned image editing tasks with a strong emphasis on correctness in action-centric dynamics. Secondly, to evaluate long-horizon world modelling, we adopt WORLD-PREDICTIONBENCH, which supports multi-step rollouts of up to four future observations across five subtasks, enabling a comprehensive assessment of FWM consistency.

To validate the generalisation of SWIRL, we conduct experiments across three textual environments on LLMs, each having a unique scheme of state transitions. We utilise SCIENCEWORLD [55] to evaluate physical dynamics, where LLM must predict the textual consequences of scientific actions within a simulated world. We also employ MIND2WEB [56], which challenges the model to forecast the updated HTML DOM tree following user interactions (e.g., clicks) on web elements. Finally, we assess functional tool dynamics via STABLETOOLBENCH [57], where the objective is to simulate the execution output of API calls conditioned on the current conversational state and the invoked tool.

Baselines. For visual world modelling, we compare SWIRL against a directly fine-tuned Liquid model (SFT). We reproduce two strong baselines from Qiu et al. [9]: TEST-TIME VERIFICATION, which selects the best sample from the fine-tuned Liquid based on IDM scores, and BOOTSTRAP, which fine-tunes Liquid on silver data annotated by the IDM. We also include specialised diffusion-based image editing models, including GoT [58], SmartEdit [59], InstructPix2Pix [60], and the Chameleon model family, following the protocol of Qiu et al. [9]. To position our approach against recent (larger or diffusion-based) unified VLMs, we also evaluate BAGEL [22], OmniGen [40], OmniGen2 [61], BLIP3o-NEXT [62], and UniWorld-V1 [39]. For text-based environments, prior work under this formulation is limited. We therefore compare against SFT baselines and larger LLMs.

Evaluation Metrics. For visual world modelling, we adopt GPT-4o-as-a-judge as the main metric for evaluation, as [9, 58] with a 10-point scheme. We also report reference-based metrics, Edit Success (ES) and CLIP scores for AURORA-BENCH for completeness, although the latter is inflated when copying the previous state as a predicted next state [49]. We report and discuss these reference-based metrics in Appendix B. For text environments, we use BLEU, BERTScore, and ROUGE-L.

Table 1: **Quantitative comparison on AURORA-BENCH.** We report GPT-4o-as-a-judge scores, EditSuccess (ES), and CLIP scores across five datasets (and their Average). Blue cells indicate where SWIRL outperforms or matches Liquid-SFT, its direct baseline.

METHOD	MAGICBRUSH			ACTION GENOME			SOMETHING			WHATSUP			KUBRIC			AVERAGE		
	GPT-4o	ES	CLIP	GPT-4o	ES	CLIP	GPT-4o	ES	CLIP	GPT-4o	ES	CLIP	GPT-4o	ES	CLIP	GPT-4o	ES	CLIP
UNIFIED VLMS																		
BAGEL-14B	8.14	0.44	0.95	6.72	0.08	0.90	6.90	0.16	0.82	5.40	0.22	0.95	5.82	0.18	0.94	6.44	0.22	0.91
OMNIGEN2	7.04	0.48	0.92	4.96	0.12	0.90	6.00	0.21	0.82	6.60	0.28	0.92	5.54	0.14	0.89	6.05	0.25	0.89
BLIP3O-NEXT	3.42	0.70	0.74	3.04	0.44	0.67	2.86	0.35	0.66	3.00	0.40	0.78	3.40	0.54	0.74	3.14	0.49	0.72
OMNIGEN	6.86	0.48	0.94	5.70	0.18	0.85	7.11	0.46	0.80	7.52	0.24	0.93	5.69	0.16	0.91	6.59	0.30	0.89
UNIWORLD-V1	7.42	0.40	0.91	7.06	0.10	0.88	7.37	0.25	0.77	8.20	0.46	0.91	6.76	0.42	0.84	7.36	0.33	0.86
EXISTING BASELINES																		
INSTRUCTPIX2PIX	3.12	—	—	1.20	—	—	0.96	—	—	0.00	—	—	1.88	—	—	1.43	—	—
GOT	5.96	—	—	1.61	—	—	2.62	—	—	1.58	—	—	3.92	—	—	3.14	—	—
SMARTEDIT	6.71	—	—	3.08	—	—	2.81	—	—	0.76	—	—	3.70	—	—	3.41	—	—
CHAMELEON-SFT	2.52	—	—	2.48	—	—	3.11	—	—	0.88	—	—	7.30	—	—	3.26	—	—
CHAMELEON-BOOTSTRAP	3.27	—	—	2.74	—	—	3.11	—	—	0.98	—	—	7.30	—	—	3.48	—	—
OUR BASELINES																		
LIQUID-SFT	5.46	0.28	0.92	2.76	0.34	0.79	3.00	0.27	0.74	3.60	0.32	0.85	7.00	0.60	0.90	4.36	0.36	0.84
LIQUID-BOOTSTRAP	5.27	0.30	0.89	3.02	0.42	0.79	2.86	0.29	0.75	3.88	0.28	0.87	5.57	0.30	0.90	4.11	0.32	0.84
LIQUID-SFT W/ TEST-TIME VERIFICATION																		
$N = 2$	5.71	0.14	0.93	3.00	0.38	0.80	3.40	0.21	0.76	4.38	0.36	0.86	6.18	0.44	0.90	4.53	0.31	0.85
$N = 4$	6.10	0.14	0.93	3.38	0.36	0.81	3.44	0.25	0.76	4.38	0.40	0.87	6.04	0.38	0.90	4.66	0.31	0.85
$N = 8$	6.18	0.16	0.92	3.28	0.36	0.81	3.74	0.21	0.74	4.48	0.28	0.86	6.28	0.46	0.90	4.77	0.29	0.85
SWIRL (IDM → FWM)	6.48	0.26	0.92	3.58	0.36	0.80	3.44	0.33	0.74	4.08	0.32	0.86	6.59	0.50	0.90	4.83	0.36	0.84
SWIRL	6.00	0.28	0.92	3.40	0.40	0.80	3.73	0.31	0.74	4.46	0.30	0.85	7.45	0.62	0.91	5.00	0.39	0.84

4.2 Open-World Visual Dynamics Prediction

AURORA-BENCH. We present the quantitative evaluation of visual world modelling on AURORA-BENCH in Table 1. The primary comparison of interest is against Liquid-SFT, our direct supervised fine-tuning baseline. As highlighted in blue, SWIRL delivers consistent improvements on GPT-4o across all five benchmarks (from 4.36 to 5.00 on average). The average Edit Success is enhanced, too, from 0.36 to 0.39. Compared to the baselines from [9], SWIRL surpasses the bootstrapping strategy using IDM to synthesise trajectories from unlabelled videos, and Test-time Verification (with up to $N = 8$ samples). Our iterative strategy also outperforms the non-iterative supervision from IDM only, i.e. SWIRL (IDM → FWM), at 4.83, indicating the effectiveness of SWIRL by alternating policy and reward roles between FWM and IDM. Furthermore, despite starting from a weaker base model, i.e., Liquid, and a lightweight post-training with 400K samples from [48, 49] to initialise the editing ability. SWIRL remains highly competitive with other state-of-the-art unified VLMs such as OmniGen2 and BAGEL, while substantially outperforming specialised diffusion-based editors like InstructPix2Pix. Qualitative examples for SWIRL are presented in Appendix M.

Table 2: **GPT4o-as-a-judge on the ByteMorph Benchmark.** Additional metrics are reported in Appendix B). Blue cells indicate where SWIRL outperforms Liquid-SFT.

METHOD	Camera Zoom	Camera Motion	Object Motion	Human Motion	Inter-action	Average
BAGEL-14B	32.89	49.34	49.33	45.03	40.89	44.48
OMNIGEN	47.76	52.17	44.88	43.59	40.92	44.40
OMNIGEN2	49.08	55.39	61.67	65.52	61.63	60.14
BLIP3O-NEXT	24.21	13.16	24.33	34.22	28.53	27.28
UNIWORLD-V1	55.53	68.03	55.91	48.10	58.03	55.56
LIQUID-SFT	57.23	56.51	43.13	38.07	40.70	43.38
SWIRL						
(IDM→FWM)	57.37	50.13	62.50	48.69	56.53	54.57
(ITERATIVE)	53.16	55.86	62.10	53.78	53.81	55.72

BYTEMORPH. As the BYTEMORPH results in Table 2 show, all variants of SWIRL substantially raise the *Average* score over the Liquid-SFT baseline, e.g. SWIRL ITERATIVE goes from 43.38 to 55.72 (+28.5%). Performance on global camera control (*Camera Zoom* and *Camera Motion*) remains comparable. This is expected since the unlabelled in-the-wild videos used during the RL phase (e.g., VIDGEN-1M) are predominantly static and provide limited supervision for camera control. In contrast, SWIRL yields pronounced improvements on *Object Motion*, *Human Motion* and *Interaction*, indicating effective learning of fine-grained dynamics. Notably, SWIRL matches the performance of larger or more heavily supervised VLMs (e.g., BAGEL and UniWorld-v1).

WORLDPREDICTIONBENCH. To assess *long-horizon* world modelling in addition to single-step prediction, we evaluate performance on WORLDPREDICTIONBENCH. The model must predict future observations autoregressively up to $T = 6$ time steps, using its own previous predictions as context. This setup tests the model's ability to maintain physical consistency and resist the compounding

Table 3: **Long-Horizon World Modelling on WorldPredictionBench.** We report GPT-4o-as-a-judge scores in the table and extra metrics in Appendix B. The left section averages performance across the horizon for each subtask (Turns 1–6 for EgoExo, EPIC, IKEA; Turns 1–4 for COIN, CrossTask). The right details the decay of the *Overall* score at each prediction turn. Blue cells indicate where SWIRL surpasses Liquid-SFT.

	Average Task Score (Horizon 1–6)						Overall Score by Turn					
Model	**COIN**†	**Cross**†	**EgoExo**	**EPIC**	**IKEA**	**Avg.**	**T=1**	**T=2**	**T=3**	**T=4**	**T=5**	**T=6**
Bagel	4.31	3.94	2.88	3.37	3.01	3.30	4.29	4.10	3.47	3.22	2.47	2.23
OmniGen2	3.78	3.53	2.73	3.01	3.24	3.28	3.25	2.64	4.05	3.75	3.06	3.11
BLIP3o-Next	1.74	1.58	1.24	1.37	1.70	1.31	2.46	1.82	1.20	0.95	0.83	0.63
OmniGen	3.61	3.48	3.20	3.14	3.56	3.45	3.25	2.83	4.00	3.92	3.28	3.45
UniWorld-v1	3.27	3.42	3.20	2.57	3.39	3.35	3.40	3.41	3.68	3.59	3.09	2.97
Liquid-SFT	2.11	2.06	1.59	1.80	1.35	1.63	3.09	2.09	1.40	1.17	1.07	0.97
SWIRL (IDM → FWM)	2.61	2.45	1.84	1.95	1.56	1.89	3.24	2.42	1.85	1.59	1.25	1.08
SWIRL (Iterative)	2.06	2.28	1.82	1.66	1.43	1.68	2.95	2.04	1.53	1.24	1.23	1.11

Table 4: **Main Results on Textual Environments.** We evaluate performance on ScienceWorld, Mind2Web, and StableToolBench. We report BERTScore (BS) and ROUGE-L (R-L) for the first two tasks, and BLEU across all tasks for StableToolBench as in [57]. **Bold** indicates the best performance in each column, blue cells indicate where SWIRL outperforms Liquid-SFT.

	ScienceWorld		Mind2Web		StableToolBench					
Model	**BS**	**R-L**	**BS**	**R-L**	**ID High**	**ID Low**	**ID Med**	**OOD**	**OOD Fail**	**Average**
Qwen-2.5-3B-Instruct	82.87	18.68	82.11	18.01	7.74	7.74	9.48	6.59	3.18	6.95
Qwen-2.5-7B-Instruct	80.48	12.50	77.03	7.35	5.93	9.89	8.30	9.17	3.20	7.30
Qwen-2.5-14B-Instruct	81.46	15.52	75.43	3.02	6.23	8.68	8.58	6.48	3.27	6.65
Qwen-2.5-32B-Instruct	81.01	14.53	79.01	11.86	7.13	7.13	10.22	7.75	4.34	7.31
Olmo-3-7B-Instruct	79.27	11.28	76.05	6.00	8.28	13.33	7.62	9.15	**7.54**	9.18
DeepSeek-7B-Chat	81.90	16.35	80.18	14.32	5.38	15.17	4.44	8.52	4.17	7.54
Qwen-2.5-3B-SFT	**96.06**	89.73	92.37	48.97	16.03	12.87	17.51	14.86	2.99	12.85
Qwen-2.5-3B-SWIRL	**96.06**	**89.92**	**92.44**	**49.10**	**16.47**	**16.90**	**21.20**	**15.57**	2.92	**14.61**

covariate shift inherent in sequential generation. Table 3 summarises the GPT-4o-as-a-judge scores across all subsets in WorldPredictionBench.

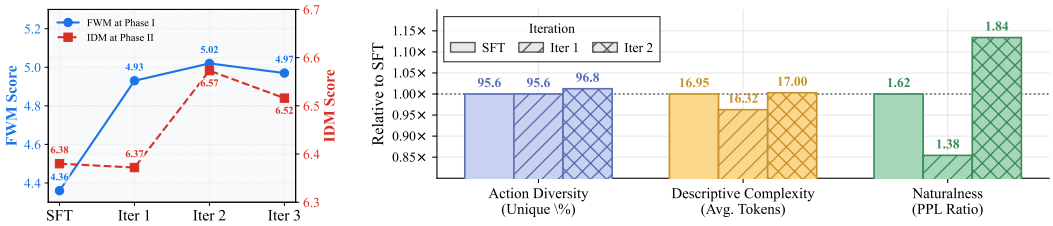
The best variant of our models demonstrates a significant improvement in temporal consistency compared to Liquid-SFT. While SFT yields competitive performance at the immediate next step ($T = 1$), it suffers from rapid degradation as the horizon increases, dropping from a score of 3.09 to 0.97 by $T = 6$. In contrast, SWIRL maintains significantly higher fidelity throughout the rollout trajectory, achieving a +14.4% relative improvement over the baseline at $T = 6$ (1.11). Notably, contrary to the other evaluation datasets, here the single-turn variant of SWIRL (IDM → FWM) surpasses the iterative one on average (1.95 versus 1.68).

4.3 Textual Environments

Table 4 shows the quantitative evaluation on textual environments. When applied to Qwen-2.5-3B-Instruct in ScienceWorld and Mind2Web, both SWIRL and SFT are near-saturation in semantic accuracy (>92 BERTScore) and comparable in exact lexical matching (ROUGE-L). The advantage of our approach becomes most pronounced in StableToolBench, which requires the simulation of API execution outcomes. Here, we observe substantial improvements in generalisation capabilities, with our method surpassing SFT by +4.03 BLEU on *ID-Low* and +3.69 BLEU on *ID-Medium* splits. This establishes a new state-of-the-art for this model scale, outperforming also significantly larger open-weight instruction models (e.g., Qwen-2.5-32B, DeepSeek-7B). Overall, this suggests that the SWIRL can generalise to internalise complex API dynamics effectively for LLMs.

4.4 Analysis

Iterative RL. We run an analysis to determine the potential for cumulative gain through iterative self-improvement beyond the first round. Figure 2a tracks the training dynamics across three iterations,



(a) Performance of FWM and IDM in each iteration of SWIRL on AURORA-BENCH.

(b) *Uniqueness* (the percentage of distinct actions in the test set), *length* (average token count), and *PPL ratio* (the ratio of GPT-2 perplexity on predicted actions to that on ground-truth actions).

Figure 2: **Iterative dynamics and latent-action analysis of SWIRL.** (a) tracks FWM and IDM evaluation scores across iterations. (b) reports the semantic properties of generated action space.

where each iteration consists of updating the FWM using the current IDM as a reward (Phase I), followed by updating the IDM using the improved FWM as a reward (Phase II). We observe a clear *virtuous cycle*: enhancing the FWM's capability enables it to provide a more robust verification for action prediction, which in turn yields a more precise reward signal for the subsequent FWM updates.

Linguistic Properties of Latent Actions. We analyse the grounded linguistic space for latent actions in Figure 2b, finding that it is robust and informative across iterations. The rate of unique actions predicted by IDM stays above 94%, the GPT-2 perplexity of predicted actions remains comparable to that of the ground truth, and the average action length (≈ 16.8 tokens) consistently exceeds the ground-truth average (≈ 7.2). We provide an in-depth discussion in Appendix G.

RL vs SFT. To isolate the contribution of the optimisation objective, we compare SWIRL against direct SFT using the same set of unlabelled videos initially annotated by our IDM model. As shown in Figure 3 with details in Appendix I, SWIRL significantly outperforms the SFT baselines (both continued training and data merging), which stagnate or degrade as data scales. We attribute this disparity to the inherent ambiguity of visual dynamics and action verbalisation. SFT enforces strict token-level imitation of the IDM's specific pseudo-labels; however, a single visual transition often corresponds to multiple valid descriptions. Forcing the model to mimic one specific verbalisation can lead to overfitting noise and suppressing valid alternative predictions. In contrast, our GRPO framework effectively relaxes this constraint by encouraging consistency assessed by IDM, rather than exact replication. By exploring the FWM's rollout space and action trajectories the IDM recognises as physically plausible, SWIRL learns a more robust and generalisable FWM that is not limited by the specific phrasing of the teacher annotations.

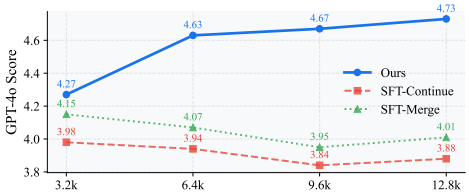


Figure 3: We compare our proposed SWIRL against SFT baselines (continual training and merging) across five benchmarks in AURORA-BENCH. The x-axis represents the number of training samples. Our method (solid blue line) demonstrates superior data efficiency, achieving higher GPT-4o evaluation scores.

5 Conclusion

We introduce SWIRL, a unified framework to post-train VLMs and LLMs and enhance their ability to model the world (future states conditioned on the current state and action), without relying on human-annotated trajectories. By interpreting actions as latent variables and alternately optimising forward world modelling and inverse dynamics modelling objectives with GRPO, our method induces self-improving world modelling purely from unlabelled data. We provide theoretical guarantees establishing the learnability of each optimisation phase, formally linking our objectives to variational mutual information bounds and evidence lower bounds. Empirically, we demonstrate the effectiveness of SWIRL across diverse settings, including real-world visual dynamics with VLMs and textual environments (physical and digital simulations and tool calling) with LLMs, validating its generality.

Impact Statement

This paper presents SWIRL, a framework that enables LLMs and VLMs to self-improve their internal world models using unlabelled data. By reducing reliance on expensive human-annotated trajectories, our work advances the efficiency and accessibility of training capable reasoning agents. This has positive implications for the development of general-purpose assistants that can better understand physical dynamics and digital environments.

However, we acknowledge potential societal consequences associated with these capabilities. First, our method utilises self-improving loops on unlabelled "in-the-wild" data. Without human curation, there is a risk that the model may internalise, reinforce, or amplify biases and harmful patterns present in the raw distribution of web and video data. Future work employing this framework should incorporate safety filters or constitutional AI principles within the reward mechanism to mitigate this risk.

Second, the improvements in Visual Dynamics Prediction and Web HTML interaction imply a step forward in generative video capabilities and autonomous web agents. While these advancements aid in creative tools and automation, they also lower the barrier for generating deepfakes or creating agents capable of bypassing web-based security measures (e.g., CAPTCHAs) or conducting automated interactions at scale. We emphasise the necessity of developing robust detection tools for synthetic media and implementing strict access controls and guardrails for autonomous agents operating in real-world web environments.

Limitations

We acknowledge several limitations of SWIRL that constrain the scope of our claims and motivate future work.

Firstly, our learnability guarantees in Theorems 3.1 and 3.2 establish that each phase corresponds to a principled variational objective (a CMI lower bound and the ELBO, respectively), but they characterise *per-phase* optimisation rather than the joint convergence of the alternating procedure. As with most coordinate-ascent schemes coupled with stochastic policy gradients, global convergence guarantees would require additional assumptions on the IDM's expressiveness, the tightness of the variational gap, and the stationarity of the empirical belief distribution $\tilde{P}(z \mid x)$ across iterations. A formal analysis of fixed points and convergence rates of the full reciprocal loop is left to future work.

Secondly, SWIRL with Liquid as a base model requires a warm-up phase to initialise image editing ability: as discussed in Appendix E, Liquid exhibits near-zero zero-shot editing ability on GEDIT-BENCH (0.03 average), and an SFT warm-up is required to bootstrap a non-trivial rollout distribution before GRPO can explore meaningfully. However, this limitation can be alleviated with a stronger base model such as BAGEL [22] or Emu3.5 [23] which can natively predict the image conditioning on the image-text input. Given the limited compute resources, we choose Liquid as our base model, which relies on curated, action-annotated editing data for initialising the correct prediction format, although the mixtures are mostly focusing on general image editing rather than world modelling, and the gains over LIQUID-SFT we report are obtained purely from observation-only sequences.

Thirdly, our self-improvement loop operates on unlabelled in-the-wild video (VIDGEN-1M), whose distribution is heavily skewed toward static scenes and object-/human-centric motion. Correspondingly, gains on BYTEMORPH are concentrated in Object Motion, Human Motion, and Interaction, while Camera Zoom and Camera Motion remain comparable to the SFT baseline. SWIRL therefore inherits the coverage biases of its unlabelled corpus, and broadening dynamics coverage will likely require targeted data curation or active sampling strategies.

References

[1] Keyon Vafa, Justin Y Chen, Ashesh Rambachan, Jon Kleinberg, and Sendhil Mullainathan. Evaluating the world model implicit in a generative model. *Advances in Neural Information Processing Systems*, 37:26941–26975, 2024.

[2] Yifu Qiu, Zheng Zhao, Yftah Ziser, Anna Korhonen, Edoardo Ponti, and Shay B Cohen. Are large language model temporally grounded? In *Proceedings of the 2024 Conference of the*

North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 7057–7076, 2024.

[3] Zeyuan Liu, Ziyu Huan, Xiyao Wang, Jiafei Lyu, Jian Tao, Xiu Li, Furong Huang, and Huazhe Xu. World models with hints of large language models for goal achieving. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 50–72, 2025.

[4] Delong Chen, Theo Moutakanni, Willy Chung, Yejin Bang, Ziwei Ji, Allen Bolourchi, and Pascale Fung. Planning with reasoning using vision language world model. *arXiv preprint arXiv:2509.02722*, 2025.

[5] Kangrui Wang, Pingyue Zhang, Zihan Wang, Yaning Gao, Linjie Li, Qineng Wang, Hanyang Chen, Yiping Lu, Zhengyuan Yang, Lijuan Wang, et al. Vagen: Reinforcing world model reasoning for multi-turn vlm agents. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

[6] Zhexiao Xiong, Xin Ye, Burhan Yaman, Sheng Cheng, Yiren Lu, Jingru Luo, Nathan Jacobs, and Liu Ren. Unidrive-wm: Unified understanding, planning and generation world model for autonomous driving. *arXiv preprint arXiv:2601.04453*, 2026.

[7] Shangmin Guo, Omar Darwiche Domingues, Raphaël Avalos, Aaron Courville, and Florian Strub. Sample, predict, then proceed: Self-verification sampling for tool use of llms. *arXiv preprint arXiv:2506.02918*, 2025.

[8] Jade Copet, Quentin Carbonneaux, Gal Cohen, Jonas Gehring, Jacob Kahn, Jannik Kossen, Felix Kreuk, Emily McMilin, Michel Meyer, Yuxiang Wei, et al. Cwm: An open-weights llm for research on code generation with world models. *arXiv preprint arXiv:2510.02387*, 2025.

[9] Yifu Qiu, Yftah Ziser, Anna Korhonen, Shay B Cohen, and Edoardo M Ponti. Bootstrapping world models from dynamics models in multimodal foundation models. *arXiv preprint arXiv:2506.06006*, 2025.

[10] Matthieu Tehenan, Christian Bolivar Moya, Tenghai Long, and Guang Lin. Linear spatial world models emerge in large language models. *arXiv preprint arXiv:2506.02996*, 2025.

[11] Emmy Liu, Varun Gangal, Chelsea Zou, Xiaoqi Huang, Michael Yu, Alex Chang, Zhuofu Tao, Sachin Kumar, and Steven Y Feng. A unified definition of hallucination, or: It's the world model, stupid. *arXiv preprint arXiv:2512.21577*, 2025.

[12] Jiongxiao Wang, Qiaojing Yan, Yawei Wang, Yijun Tian, Soumya Smruti Mishra, Zhichao Xu, Megha Gandhi, Panpan Xu, and Lin Lee Cheong. Reinforcement learning for self-improving agent with skill library. *arXiv preprint arXiv:2512.17102*, 2025.

[13] Yinjie Wang, Ling Yang, Ye Tian, Ke Shen, and Mengdi Wang. Cure: Co-evolving coders and unit testers via reinforcement learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[14] Weiyang Jin, Yuwei Niu, Jiaqi Liao, Chengqi Duan, Aoxue Li, Shenghua Gao, and Xihui Liu. Srum: Fine-grained self-rewarding for unified multimodal models. *arXiv preprint arXiv:2510.12784*, 2025.

[15] Weijia Mao, Zhenheng Yang, and Mike Zheng Shou. Unirl: Self-improving unified multimodal models via supervised and reinforcement learning. *arXiv preprint arXiv:2505.23380*, 2025.

[16] 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 *Forty-first International Conference on Machine Learning*, 2024.

[17] Yucen Wang, Fengming Zhang, De-Chuan Zhan, Li Zhao, Kaixin Wang, and Jiang Bian. Co-evolving latent action world models. *arXiv preprint arXiv:2510.26433*, 2025.

[18] Qineng Wang, Wenlong Huang, Yu Zhou, Hang Yin, Tianwei Bao, Jianwen Lyu, Weiyu Liu, Ruohan Zhang, Jiajun Wu, Li Fei-Fei, et al. Enact: Evaluating embodied cognition with world modeling of egocentric interaction. *arXiv preprint arXiv:2511.20937*, 2025.

[19] Delong Chen, Willy Chung, Yejin Bang, Ziwei Ji, and Pascale Fung. Worldprediction: A benchmark for high-level world modeling and long-horizon procedural planning. *arXiv preprint arXiv:2506.04363*, 2025.

[20] David Barber and Felix Agakov. The im algorithm: a variational approach to information maximization. In *Proceedings of the 17th International Conference on Neural Information Processing Systems*, NIPS'03, page 201–208, Cambridge, MA, USA, 2003. MIT Press.

[21] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[22] Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, et al. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025.

[23] Yufeng Cui, Honghao Chen, Haoge Deng, Xu Huang, Xinghang Li, Jirong Liu, Yang Liu, Zhuoyan Luo, Jinsheng Wang, Wenxuan Wang, et al. Emu3. 5: Native multimodal models are world learners. *arXiv preprint arXiv:2510.26583*, 2025.

[24] Wolfgang Lehrach, Daniel Hennes, Miguel Lazaro-Gredilla, Xinghua Lou, Carter Wendelken, Zun Li, Antoine Dedieu, Jordi Grau-Moya, Marc Lanctot, Atil Iscen, et al. Code world models for general game playing. *arXiv preprint arXiv:2510.04542*, 2025.

[25] Yixia Li, Hongru Wang, Jiahao Qiu, Zhenfei Yin, Dongdong Zhang, Cheng Qian, Zeping Li, Pony Ma, Guanhua Chen, Heng Ji, et al. From word to world: Can large language models be implicit text-based world models? *arXiv preprint arXiv:2512.18832*, 2025.

[26] Kaige Xie, Ian Yang, John Gunerli, and Mark Riedl. Making large language models into world models with precondition and effect knowledge. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 7532–7545, 2025.

[27] 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.

[28] Qiyue Gao, Xinyu Pi, Kevin Liu, Junrong Chen, Ruolan Yang, Xinqi Huang, Xinyu Fang, Lu Sun, Gautham Kishore, Bo Ai, et al. Do vision-language models have internal world models? towards an atomic evaluation. *arXiv preprint arXiv:2506.21876*, 2025.

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

[30] Hongzhe Bi, Hengkai Tan, Shenghao Xie, Zeyuan Wang, Shuhe Huang, Haitian Liu, Ruowen Zhao, Yao Feng, Chendong Xiang, Yinze Rong, et al. Motus: A unified latent action world model. *arXiv preprint arXiv:2512.13030*, 2025.

[31] Quentin Garrido, Tushar Nagarajan, Basile Terver, Nicolas Ballas, Yann LeCun, and Michael Rabbat. Learning latent action world models in the wild. *arXiv preprint arXiv:2601.05230*, 2026.

[32] Erik Bauer, Elvis Nava, and Robert K Katzschmann. Latent action diffusion for cross-embodiment manipulation. *arXiv preprint arXiv:2506.14608*, 2025.

[33] Hamza Adnan, Matthew T Jackson, and Alexey Zakharov. Segment to focus: Guiding latent action models in the presence of distractors. *arXiv preprint arXiv:2602.02259*, 2026.

[34] Jiaxin Huang, Shixiang Gu, Le Hou, Yuexin Wu, Xuezhi Wang, Hongkun Yu, and Jiawei Han. Large language models can self-improve. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 1051–1068, 2023.

[35] Shelly Bensal, Umar Jamil, Christopher Bryant, Melisa Russak, Kiran Kamble, Dmytro Mozolevskyi, Muayad Ali, and Waseem AlShikh. Reflect, retry, reward: Self-improving llms via reinforcement learning. *arXiv preprint arXiv:2505.24726*, 2025.

[36] Nayoung Lee, Ziyang Cai, Avi Schwarzschild, Kangwook Lee, and Dimitris Papailiopoulos. Self-improving transformers overcome easy-to-hard and length generalization challenges. In *Forty-second International Conference on Machine Learning*.

[37] Chenyang Zhao, Xueying Jia, Vijay Viswanathan, Tongshuang Wu, and Graham Neubig. Self-guide: Better task-specific instruction following via self-synthetic finetuning. *arXiv preprint arXiv:2407.12874*, 2024.

[38] Junfeng Wu, Yi Jiang, Chuofan Ma, Yuliang Liu, Hengshuang Zhao, Zehuan Yuan, Song Bai, and Xiang Bai. Liquid: Language models are scalable and unified multi-modal generators. *arXiv preprint arXiv:2412.04332*, 2024.

[39] Bin Lin, Zongjian Li, Xinhua Cheng, Yuwei Niu, Yang Ye, Xianyi He, Shenghai Yuan, Wangbo Yu, Shaodong Wang, Yunyang Ge, et al. Uniworld: High-resolution semantic encoders for unified visual understanding and generation. *arXiv preprint arXiv:2506.03147*, 2025.

[40] Shitao Xiao, Yueze Wang, Junjie Zhou, Huaying Yuan, Xingrun Xing, Ruiran Yan, Chaofan Li, Shuting Wang, Tiejun Huang, and Zheng Liu. Omnigen: Unified image generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 13294–13304, 2025.

[41] Yang Shi, Yuhao Dong, Yue Ding, Yuran Wang, Xuanyu Zhu, Sheng Zhou, Wenting Liu, Haochen Tian, Rundong Wang, Huanqian Wang, et al. Realunify: Do unified models truly benefit from unification? a comprehensive benchmark. *arXiv preprint arXiv:2509.24897*, 2025.

[42] Chuofan Ma, Yi Jiang, Junfeng Wu, Jihan Yang, Xin Yu, Zehuan Yuan, Bingyue Peng, and Xiaojuan Qi. Unitok: A unified tokenizer for visual generation and understanding. *arXiv preprint arXiv:2502.20321*, 2025.

[43] Liao Qu, Huichao Zhang, Yiheng Liu, Xu Wang, Yi Jiang, Yiming Gao, Hu Ye, Daniel K Du, Zehuan Yuan, and Xinglong Wu. Tokenflow: Unified image tokenizer for multimodal understanding and generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 2545–2555, 2025.

[44] Dian Zheng, Manyuan Zhang, Hongyu Li, Kai Zou, Hongbo Liu, Ziyu Guo, Kaituo Feng, Yexin Liu, Ying Luo, Yan Feng, et al. Architecture decoupling is not all you need for unified multimodal model. *arXiv preprint arXiv:2511.22663*, 2025.

[45] Jihai Zhang, Tianle Li, Linjie Li, Zhengyuan Yang, and Yu Cheng. Are unified vision-language models necessary: Generalization across understanding and generation. *arXiv preprint arXiv:2505.23043*, 2025.

[46] Xinyu Qiu, Heng Jia, Zhengwen Zeng, Shuheng Shen, Changhua Meng, Yi Yang, and Linchao Zhu. Unified generation and self-verification for vision-language models via advantage decoupled preference optimization. *arXiv preprint arXiv:2601.01483*, 2026.

[47] Qwen Team. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

[48] Yusu Qian, Eli Bocek-Rivele, Liangchen Song, Jialing Tong, Yinfei Yang, Jiasen Lu, Wenze Hu, and Zhe Gan. Pico-banana-400k: A large-scale dataset for text-guided image editing. *arXiv preprint arXiv:2510.19808*, 2025.

[49] Benno Krojer, Dheeraj Vattikonda, Luis Lara, Varun Jampani, Eva Portelance, Christopher Pal, and Siva Reddy. Learning Action and Reasoning-Centric Image Editing from Videos and Simulations. In *NeurIPS*, 2024. URL `https://arxiv.org/abs/2407.03471`. Spotlight Paper.

[50] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[51] Mathew Monfort, Alex Andonian, Bolei Zhou, Kandan Ramakrishnan, Sarah Adel Bargal, Tom Yan, Lisa Brown, Quanfu Fan, Dan Gutfruend, Carl Vondrick, et al. Moments in time dataset: one million videos for event understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–8, 2019. ISSN 0162-8828. doi: 10.1109/TPAMI.2019.2901464.

[52] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

[53] Zhiyu Tan, Xiaomeng Yang, Luozheng Qin, and Hao Li. Vidgen-1m: A large-scale dataset for text-to-video generation. *arXiv preprint arXiv:2408.02629*, 2024.

[54] Xi Chen, Zhifei Zhang, He Zhang, Yuqian Zhou, Soo Ye Kim, Qing Liu, Yijun Li, Jianming Zhang, Nanxuan Zhao, Yilin Wang, et al. Unireal: Universal image generation and editing via learning real-world dynamics. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 12501–12511, 2025.

[55] Ruoyao Wang, Peter Jansen, Marc-Alexandre Côté, and Prithviraj Ammanabrolu. Scienceworld: Is your agent smarter than a 5th grader? *arXiv preprint arXiv:2203.07540*, 2022.

[56] Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Sam Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2web: Towards a generalist agent for the web. *Advances in Neural Information Processing Systems*, 36:28091–28114, 2023.

[57] Zhicheng Guo, Sijie Cheng, Hao Wang, Shihao Liang, Yujia Qin, Peng Li, Zhiyuan Liu, Maosong Sun, and Yang Liu. Stabletoolbench: Towards stable large-scale benchmarking on tool learning of large language models. *arXiv preprint arXiv:2403.07714*, 2024.

[58] Rongyao Fang, Chengqi Duan, Kun Wang, Linjiang Huang, Hao Li, Shilin Yan, Hao Tian, Xingyu Zeng, Rui Zhao, Jifeng Dai, et al. GoT: Unleashing reasoning capability of multimodal large language model for visual generation and editing. *arXiv preprint arXiv:2503.10639*, 2025.

[59] Yuzhou Huang, Liangbin Xie, Xintao Wang, Ziyang Yuan, Xiaodong Cun, Yixiao Ge, Jiantao Zhou, Chao Dong, Rui Huang, Ruimao Zhang, et al. SmartEdit: Exploring complex instruction-based image editing with multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8362–8371, 2024.

[60] Tim Brooks, Aleksander Holynski, and Alexei A Efros. InstructPix2Pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18392–18402, 2023.

[61] Chenyuan Wu, Pengfei Zheng, Ruiran Yan, Shitao Xiao, Xin Luo, Yueze Wang, Wanli Li, Xiyan Jiang, Yexin Liu, Junjie Zhou, Ze Liu, Ziyi Xia, Chaofan Li, Haoge Deng, Jiahao Wang, Kun Luo, Bo Zhang, Defu Lian, Xinlong Wang, Zhongyuan Wang, Tiejun Huang, and Zheng Liu. Omnigen2: Exploration to advanced multimodal generation. *arXiv preprint arXiv:2506.18871*, 2025.

[62] Jiuhai Chen, Le Xue, Zhiyang Xu, Xichen Pan, Shusheng Yang, Can Qin, An Yan, Honglu Zhou, Zeyuan Chen, Lifu Huang, et al. Blip3o-next: Next frontier of native image generation. *arXiv preprint arXiv:2510.15857*, 2025.

[63] Di Chang, Mingdeng Cao, Yichun Shi, Bo Liu, Shengqu Cai, Shijie Zhou, Weilin Huang, Gordon Wetzstein, Mohammad Soleymani, and Peng Wang. Bytemorph: Benchmarking instruction-guided image editing with non-rigid motions. *arXiv preprint arXiv:2506.03107*, 2025.

[64] Joao Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. A short note on the kinetics-700 human action dataset. *arXiv preprint arXiv:1907.06987*, 2019.

[65] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In Qun Liu and David Schlangen, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-demos.6. URL `https://aclanthology.org/2020.emnlp-demos.6/`.

A Theoretical Derivation

In this section, we provide the detailed theoretical justification for SWIRL. We formalise the interaction between the Forward World Modelling (FWM) and Inverse Dynamics Model (IDM) as an alternating maximisation of *Identifiability* (via Variational Mutual Information) and *Data Fidelity* (via the Evidence Lower Bound).

We define the following notation:

- $x \in \mathcal{S}$: The source observation (current state s_t).
- $y \in \mathcal{S}$: The ground-truth target observation (next state s_{t+1}), distributed according to the data distribution $\mathcal{D}(y|x)$.
- $\hat{y} \in \mathcal{S}$: A generated target observation sampled from the forward model.
- $z \in \mathcal{A}$: The latent action driving the transition.

We employ two parametrised models:

1. **Forward World Modelling (FWM):** $P_\theta(\hat{y}|x, z)$, parametrised by θ. FWM parametrised the environment's transition distribution.
2. **Inverse Dynamics Model (IDM):** $Q_\phi(z|x, y)$, parametrised by ϕ. IDM parametrised an approximate posterior over latent actions given a state transition.

A.1 Phase I: Learnability of the Forward Model (FWM)

In the first phase, we freeze the IDM parameters ϕ and optimise the FWM parameters θ. Our objective is to generate trajectories $\hat{y}$ that are *identifiable* by the inference model. We formalise this as maximising the **Conditional Mutual Information** (CMI) between the latent action Z and the generated observation $\hat{Y}$, conditioned on the source state X.

Crucially, in our algorithm, the latent actions Z used for training are not sampled from a fixed uninformative prior, but are inferred from the ground-truth data using the current IDM. We denote this *Empirical Belief Distribution* as $\tilde{P}(z|x)$:

$$\tilde{P}(z|x) \triangleq \mathbb{E}_{y\sim\mathcal{D}(y|x)} \left[Q_\phi(z|x, y)\right] \tag{5}$$

Theorem A.1. *Optimising the FWM to maximise the likelihood assigned by the frozen IDM to generated samples maximises a Variational Lower Bound of the Conditional Mutual Information $I_{\tilde{P}}(Z; \hat{Y}|X)$ defined over the empirical belief distribution.*

Proof. The Conditional Mutual Information under the joint distribution induced by the data and the frozen IDM is:

$$I_{\tilde{P}}(Z; \hat{Y}|X) = H_{\tilde{P}}(Z|X) - H(Z|\hat{Y}, X) \tag{6}$$

The term $H_{\tilde{P}}(Z|X)$ is the entropy of the marginal distribution of actions inferred from the dataset. Since ϕ is frozen in this phase and the dataset $\mathcal{D}$ is fixed, $\tilde{P}(z|x)$ is constant with respect to θ. Therefore, maximising the mutual information is equivalent to minimizing the conditional entropy $H(Z|\hat{Y}, X)$.

The conditional entropy is defined as:

$$H(Z|\hat{Y}, X) = -\mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{z\sim\tilde{P}(z|x)}\mathbb{E}_{\hat{y}\sim P_\theta(\cdot|x,z)} \left[\log P(z|\hat{y}, x)\right] \tag{7}$$

The true posterior $P(z|\hat{y}, x)$ is intractable as it requires marginalising over the action space. We utilise the frozen IDM $Q_\phi(z|x, \hat{y})$ as a variational approximation. By the non-negativity of the KL divergence $D_{\mathrm{KL}}(P(z|\hat{y}, x)||Q_\phi(z|x, \hat{y})) \geq 0$, we have the lower bound:

$$\mathbb{E}_{\hat{y}}[\log P(z|\hat{y}, x)] \geq \mathbb{E}_{\hat{y}}[\log Q_\phi(z|x, \hat{y})] \tag{8}$$

Substituting this into the entropy term yields the variational lower bound for the CMI:

$$I_{\tilde{P}}(Z; \hat{Y}|X) \geq H_{\tilde{P}}(Z|X) + \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{z\sim\tilde{P}(z|x)}\mathbb{E}_{\hat{y}\sim P_\theta(\cdot|x,z)} \left[\log Q_\phi(z|x, \hat{y})\right] \tag{9}$$

Expanding the definition of $\tilde{P}(z|x)$ from Eq. (1), the optimization objective becomes:

$$\mathcal{J}_{\text{FWM}}(\theta) = \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y\sim\mathcal{D}(y|x)}\left[\mathbb{E}_{z\sim Q_\phi(\cdot|x,y)}\left[\mathbb{E}_{\hat{y}\sim P_\theta(\cdot|x,z)}\left[\log Q_\phi(z|x,\hat{y})\right]\right]\right] \tag{10}$$

□

A.2 Phase II: Learnability of the Inverse Model (IDM)

In the second phase, we freeze θ and optimise the IDM parameters ϕ. We seek to maximise the log-likelihood of the observed ground-truth dynamics $\log P_\theta(y|x)$ (Data Fidelity). We model this via the Evidence Lower Bound (ELBO), treating the initialised policy at the start of the iteration as the reference prior, denoted $\pi_{\text{ref}}(z|x)$.

Theorem A.2. *Optimising the IDM via the KL-regularised policy gradient objective with reward $R(x,z,y) = \log P_\theta(y|x,z)$, $\beta = 1$, and reference policy π_{ref} maximises the Evidence Lower Bound (ELBO).*

Proof. We express the marginal log-likelihood of the data by introducing the variational distribution $Q_\phi(z|x,y)$:

$$\begin{aligned}\log P_\theta(y|x) &= \log \sum_{z\in\mathcal{A}} P_\theta(y|x,z)\pi_{\text{ref}}(z|x) \\ &= \log \mathbb{E}_{z\sim Q_\phi(\cdot|x,y)}\left[\frac{P_\theta(y|x,z)\pi_{\text{ref}}(z|x)}{Q_\phi(z|x,y)}\right]\end{aligned} \tag{11}$$

Applying Jensen's inequality (concavity of log):

$$\begin{aligned}\log P_\theta(y|x) &\geq \mathbb{E}_{z\sim Q_\phi}\left[\log P_\theta(y|x,z) + \log \pi_{\text{ref}}(z|x) - \log Q_\phi(z|x,y)\right] \\ &= \mathbb{E}_{z\sim Q_\phi}\left[\log P_\theta(y|x,z)\right] - D_{\text{KL}}(Q_\phi(z|x,y)\,||\,\pi_{\text{ref}}(z|x))\end{aligned} \tag{12}$$

This is the standard Evidence Lower Bound. The KL-regularised policy gradient objective (Phase II) is defined as:

$$\mathcal{J}(\phi) = \mathbb{E}_{z\sim Q_\phi(\cdot|x,y)}\left[R(x,z,y)\right] - \beta\, D_{\text{KL}}(Q_\phi(\cdot|x,y)\,||\,\pi_{\text{ref}}(\cdot|x)) \tag{13}$$

By setting the reward to the FWM log-likelihood, $R = \log P_\theta(y|x,z)$, we observe that:

$$\mathcal{J}(\phi) \equiv \text{ELBO}(\phi) \tag{14}$$

Thus, the IDM update step performs coordinate ascent on the evidence lower bound of the data likelihood, ensuring the inferred actions explain the ground-truth transitions under the current forward dynamics. □

B Additional Metrics

We adopt GPT-4o-as-a-judge as our primary evaluation metric throughout the main paper because reference-free evaluation is more reliable for world modelling, given its inherent ambiguity [9, 49]: a single source observation paired with a latent action admits multiple equally valid target observations, which penalises pixel-aligned reference-based metrics. Nevertheless, to provide a more comprehensive view of model behaviour, we complement the GPT-4o judgements with reference-based metrics. Specifically, we report *Editing Success* and CLIP scores on BYTEMORPH (following the AURORA-BENCH protocol), and *SSIM* and *LPIPS* as multi-frame perceptual metrics on WORLDPREDICTIONBENCH.

ByteMorph. Table 5 reports the additional metrics on BYTEMORPH. We observe that LIQUID-SFT and the SWIRL variants achieve comparable CLIP scores ($\sim$0.86–0.88), whereas the *Editing Success* score of SWIRL is substantially higher than that of LIQUID-SFT (e.g., 0.18 vs. 0.09 for SWIRL). This pattern is consistent with the observation in Krojer et al. [49]: SFT models tend to copy the source observation directly as the target prediction, which inflates the CLIP score (since the source and target are already visually similar) without actually performing the requested transition. The Editing Success metric, which explicitly rewards faithful execution of the action, exposes this failure mode and confirms that SWIRL learns to apply non-trivial dynamics rather than relying on identity shortcuts.

Table 5: Additional reference-based metrics on BYTEMORPH. CLIP alone is misleading because SFT exploits source-copying, which inflates similarity. Editing Success reveals that SWIRL variants substantially outperform LIQUID-SFT in actually performing the requested transitions.

Model	GPT Score	Edit Success	CLIP
LIQUID-SFT	40.86	0.09	**0.88**
SWIRL (IDM→FWM)	53.21	0.15	0.87
SWIRL	**54.48**	**0.18**	0.86

WorldPredictionBench. Tables 6 and 7 report SSIM (↑) and LPIPS (↓) on WORLDPREDICTION-BENCH, broken down both by subtask and by prediction turn $T = 1, \ldots, 6$. All SWIRL variants outperform LIQUID-SFT on both metrics on average, with SWIRL (IDM→FWM) achieving the best overall LPIPS (0.6671 vs. 0.6747 for SFT). Importantly, the trend is consistent with the GPT-4o-as-a-judge results in the main paper (Table 3): the gap between SWIRL and LIQUID-SFT *widens as the prediction horizon grows*. For example, at $T = 6$, SWIRL reaches an SSIM of 0.3974 vs. 0.3955 for LIQUID-SFT, and an LPIPS of 0.682 vs. 0.7015. This demonstrates that the advantage of SWIRL is most pronounced for long-horizon world modelling, where compounding covariate shift is the dominant failure mode for SFT and where reciprocal self-improvement provides the strongest inductive bias.

Table 6: SSIM (↑) on WORLDPREDICTIONBENCH, averaged across subtasks (left) and broken down by prediction turn (right). Higher is better.

Model	COIN	Cross	EgoExo	EPIC	IKEA	Avg.	T=1	T=2	T=3	T=4	T=5	T=6
LIQUID-SFT	**0.4505**	0.3952	**0.3706**	0.3230	0.4609	0.4001	**0.4571**	0.4234	0.4148	0.4058	0.3953	0.3955
SWIRL (IDM→FWM)	0.4457	**0.3954**	0.3705	0.3259	**0.4634**	0.4002	0.4530	0.4230	0.4149	0.4076	0.3997	0.3965
SWIRL	0.4465	0.3952	0.3697	**0.3320**	0.4618	**0.4010**	0.4517	**0.4244**	**0.4163**	0.4077	**0.3998**	**0.3974**

Table 7: LPIPS (↓) on WORLDPREDICTIONBENCH, averaged across subtasks (left) and broken down by prediction turn (right). Lower is better.

Model	COIN	Cross	EgoExo	EPIC	IKEA	Avg.	T=1	T=2	T=3	T=4	T=5	T=6
LIQUID-SFT	0.7042	0.6935	0.6245	0.7021	0.6491	0.6747	**0.5753**	0.6748	0.7007	0.6996	0.6924	0.7015
SWIRL (IDM→FWM)	**0.6989**	**0.6877**	0.6133	**0.6985**	0.6372	**0.6671**	0.5849	**0.6681**	**0.6905**	0.6858	0.6750	0.6848
SWIRL	0.7037	0.6921	**0.6129**	0.6994	0.6358	0.6688	0.5891	0.6708	0.6916	0.6855	0.6734	0.6820

Taken together, these additional metrics corroborate the conclusions drawn from the GPT-4o evaluations in the main paper: SWIRL produces forward predictions that are not only judged as semantically more faithful by a strong reference-free evaluator, but also align more closely with reference observations in terms of structural similarity and learned perceptual distance, particularly under the long-horizon regime.

C Sensitivity Analysis for GPT-as-a-Judge Evaluation

In this section, we repeat the GPT-4o judging procedure three independent times for each model on AURORA-BENCH, BYTEMORPH, and WORLDPREDICTIONBENCH, and report the mean and standard deviation of the resulting scores. This protocol isolates the variance attributable to the judge itself from the variance of the underlying models.

Tables 8, 9, and 10 summarise the results. Across all three benchmarks, the gap between the SWIRL variants and the LIQUID-SFT baseline substantially exceeds the standard deviation of the macro-averaged scores. For example, on BYTEMORPH, the *Overall* score increases from 40.86 ± 0.50 for LIQUID-SFT to 54.48 ± 1.03 for SWIRL, with a gap of more than 12 points against a combined standard deviation of under 1. Analogous margins hold on AURORA-BENCH (e.g., 4.68 ± 0.11 vs. 5.02 ± 0.10 on average) and on WORLDPREDICTIONBENCH (e.g., 1.79 ± 0.04 vs. 1.96 ± 0.04 on average). The improvements reported in the main paper are therefore statistically meaningful rather than artefacts of evaluator variance.

Table 8: Sensitivity analysis on AURORA-BENCH. Mean $\pm$ standard deviation of GPT-4o-as-a-judge scores over three independent evaluation runs. MB: MagicBrush, AG: Action Genome, ST: Something, WU: Whatsup, KU: Kubric.

Method	MB	AG	ST	WU	KU	Avg.
LIQUID-SFT	6.10 ± 0.07	3.20 ± 0.18	3.57 ± 0.20	4.53 ± 0.06	6.04 ± 0.31	4.68 ± 0.11
SWIRL (IDM→FWM)	6.33 ± 0.29	3.40 ± 0.17	$\mathbf{3.95 \pm 0.07}$	4.29 ± 0.20	6.68 ± 0.25	4.93 ± 0.15
SWIRL	6.40 ± 0.34	3.23 ± 0.18	3.82 ± 0.31	$\mathbf{4.56 \pm 0.03}$	$\mathbf{7.09 \pm 0.25}$	$\mathbf{5.02 \pm 0.19}$

Table 9: Sensitivity analysis on BYTEMORPH. Mean $\pm$ standard deviation of GPT-4o-as-a-judge scores over three independent evaluation runs.

Method	Camera Zoom	Camera Motion	Object Motion	Human Motion	Interaction	Overall
LIQUID-SFT	50.83 ± 3.86	50.28 ± 1.81	42.13 ± 0.58	35.37 ± 0.24	39.74 ± 1.00	40.86 ± 0.50
SWIRL (IDM→FWM)	$\mathbf{54.96 \pm 1.81}$	53.33 ± 3.60	59.38 ± 3.26	47.19 ± 1.07	$\mathbf{54.87 \pm 1.24}$	53.21 ± 1.36
SWIRL	53.47 ± 3.92	54.47 ± 2.00	$\mathbf{61.03 \pm 3.53}$	$\mathbf{51.65 \pm 1.53}$	53.13 ± 0.90	$\mathbf{54.48 \pm 1.03}$

D Implementation Details

D.1 Hyperparameters

For the LIQUID-SFT model, we fine-tune Liquid-7B [38] on PICO-BANANA-400K [48] and AURORA's training set. Training is performed for 5 epochs with a batch size of 128. We use a learning rate of 2×10^{-5} with a cosine learning rate schedule, allocating 5% of the total training steps for warm-up.

For the non-iterative setup (SWIRL (IDM $\rightarrow$ FWM)), the model is trained with a batch size of 64 and a learning rate of 1×10^{-6}. A cosine learning rate schedule with 100 warm-up steps is applied. GRPO is used with a rollout size of 8 and a KL regularization coefficient $\beta = 0.1$.

In the iterative SWIRL variant, the learning rate is increased to 5×10^{-7} while all other settings remain unchanged. In both iterative GRPO setups, we set the decoding temperature to 0.75 and Top-P to 0.96 to control the quality of predicted futures. Additionally, we apply a logit processor to constrain Liquid's generation to image tokens for FWM and textual tokens for IDM.

For LLM experiments on STABLETOOLBENCH, we fine-tune models using GRPO with DeepSpeed for memory-efficient distributed training. Optimization is performed with a learning rate of 5×10^{-5} under a cosine learning rate schedule with 25 warm-up steps, using an effective batch size of 128. For each prompt, we sample 64 rollouts. The maximum prompt and completion lengths are set to 8,126 and 4,096 tokens, respectively. Unless otherwise specified, we use a KL regularization coefficient of $\beta = 0.1$. During GRPO rollout, we set the decoding temperature to 0.7 and Top-P to 0.96. For MIND2WEB and SCIENCEWORLD, we adopt the same configuration, except that the maximum prompt and completion lengths are both set to 4,096 tokens, and the number of GRPO rollouts is reduced to 16.

For VLM experiments, training is conducted using DeepSpeed on 32 NVIDIA H200 140GB GPUs for FWM and 8 NVIDIA H200 140GB GPUs for IDM. For LLM experiments, both FWM and IDM are trained on 8 NVIDIA Grace-Hopper (GH100) GPUs.

D.2 Evaluation Prompt for GPT-4o as a Judge

For AURORA-BENCH, we use the same prompt as in Deng et al. [22]. For BYTEMORPH, we employ the official GPT-4o judge template provided by Chang et al. [63]. Since WORLDPREDICTIONBENCH [19] does not natively support multi-turn image editing evaluation, we adopt the prompt template shown in Figure 4. This template is designed to balance the penalties for directly copying the source image and for excessive editing.

Table 10: Sensitivity analysis on WORLDPREDICTIONBENCH. Mean ± standard deviation of GPT-4o-as-a-judge scores over three independent evaluation runs, broken down by prediction turn T.

Method	Avg.	T=1	T=2	T=3	T=4	T=5	T=6
LIQUID-SFT	1.79 ± 0.04	3.12 ± 0.02	1.99 ± 0.08	1.41 ± 0.02	1.19 ± 0.03	1.09 ± 0.07	0.97 ± 0.05
SWIRL (IDM→FWM)	**1.96 ± 0.04**	**3.23 ± 0.04**	**2.15 ± 0.05**	**1.57 ± 0.04**	**1.37 ± 0.03**	1.20 ± 0.04	**1.10 ± 0.05**
SWIRL	1.86 ± 0.03	2.99 ± 0.06	2.04 ± 0.02	1.55 ± 0.03	1.34 ± 0.08	**1.24 ± 0.02**	**1.10 ± 0.01**

WorldPrediction GPT-4o-as-a-Judge Prompt

You are evaluating predicted observations for a procedural plan. The plan belongs to dataset `{dataset}`, sample `{sample_uid}`. Overall plan: `{plan_description}`. Focus on step `{step_index}` of `{total_steps}`, where the action is "`{action_label}`".

You are given the source observation (input to the model) and the candidate observation (model output). There is NO ground-truth reference image.

Your task is to judge whether the candidate correctly applies the stated action to the source. Do NOT score based on overall visual quality or realism.

Evaluation rules:

1. Infer what *must change* and what *must remain unchanged* given the action.
2. Compare candidate against source to check whether the required change is present.
3. If the candidate is identical or near-identical to the source when the action implies a visible change, assign a low score.
4. Penalise unintended changes outside the scope of the action.

Return a JSON object like `{"score": <0-10>, "explanation": "..."}` summarising correctness.

Figure 4: Prompt template used to instruct GPT-4o to evaluate multi-turn visual dynamics prediction in the WorldPrediction benchmark.

E Sanity Check Results for General Image Editing.

Table 11 reports a sanity-check evaluation of general image editing performance on GEDIT-BENCH. Although Liquid does not have native image editing support by design, we find that after our supervised fine-tuning (SFT) pipeline, using a mixture of AURORA [49] and PICO-BANANA-400K [48], the resulting model (Liquid-SFT) acquires functional image editing behaviour. We illustrate the qualitative examples for general image editing in Figure 5.

Concretely, Liquid without SFT indicates a totally failure in GEDIT-BENCH, indicating the necessarily in the warm-up stage before SWIRL. Liquid-SFT achieves non-trivial scores in both semantic alignment and perceptual quality, confirming that the model can follow image editing instructions and produce coherent visual outputs. While Liquid-SFT remains significantly behind state-of-the-art proprietary and public image editing models, this result is not intended to be competitive; rather, it serves as a capability verification that our SFT pipeline successfully activates basic visual editing skills in a base model not originally designed for this task.

We emphasise that this capability plays an important role as a warm-up stage for subsequent world modelling objectives. In later stages, the model is trained to (i) predict the next visual observation given the current state (forward world modelling), and (ii) perform inverse dynamics modelling (IDM), where the model receives a pair of observations and predicts the intervening action in language. The ability to perform instruction-followed image transformations provides a minimal but necessary foundation for these more structured visual–temporal reasoning tasks.

Finally, the ablation study highlights the importance of PICO-BANANA-400K in our data mixture. Removing this component leads to a consistent degradation across all metrics (average score dropping from 3.06 to 2.43), indicating that Pico-banana provides critical signal for stabilising and enriching visual instruction-following behaviour. This result suggests that even for non-native capabilities, carefully curated visual editing data is essential for preserving functional performance after SFT.

We include the qualitative examples produced by our approach for *general image editing* in Figure 5.

Table 11: General image editing performance on GEDIT-BENCH. As a sanity check, we evaluate the image editing performance of our SFT baseline (*Liquid-SFT*) against state-of-the-art image editing models. We report Semantics, Quality, and the Average score. Results indicate that our model maintains functional editing capabilities after our SFT pipeline.

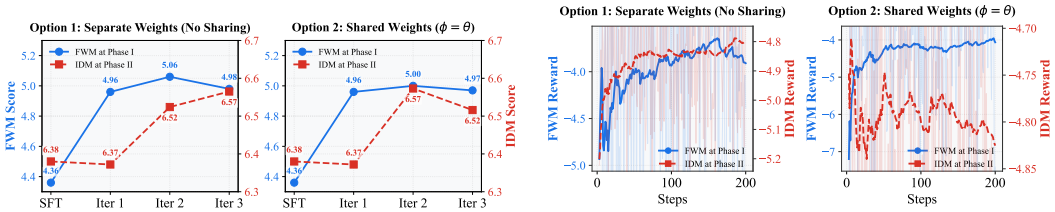
Models	Semantics	Quality	Average
Proprietary			
GPT-4o	7.85	**7.62**	**7.53**
Gemini 2.0	6.73	6.61	6.32
Public VLMs			
Emu3.5-33B	**8.27**	7.28	7.53
BAGEL-14B	7.36	6.83	6.52
OmniGen2	6.35	6.03	5.57
OmniGen-v1-diffusers	5.51	5.37	4.62
UniWorld-V1	5.05	6.85	4.80
BLIP3o-NEXT-edit-VAE	3.33	5.56	3.73
Liquid Zero-shot	0.04	1.43	0.03
Liquid-SFT	4.04	3.04	3.06
Liquid-SFT w/o PICO-BANANA-400K	2.92	2.87	2.43

Figure 5: Qualitative of SWIRL for general image editing including Subject's add/replace/remove, style transfer, altering the colour or background.

Table 12: **AURORA-BENCH Evaluation of IDM.** We compare our proposed method against baselines on captioning/description quality. We report GPT-4 evaluation scores and reference-based metrics (BERTScore, ROUGE-L, BLEU). Blue cells indicate where ours positively improve Liquid-SFT.

Method	GPT4o	BERT	R-L	BLEU
OmniGen2	6.15	14.5	15.3	7.5
Chameleon	—	**45.0**	**44.0**	20.0
Liquid-AURORA	4.52	31.0	31.0	**28.0**
Liquid-SFT	6.38	41.0	36.0	21.0
SWIRL (FWM → IDM)	6.38	40.1	39.2	20.3
SWIRL (Iter. + Share)	6.57	40.3	39.1	21.1

F Ablation in Sharing θ and ϕ



(a) Evaluation performance across iterations.

(b) Training reward curves.

Figure 6: Performance of FWM and IDM in each iteration of SWIRL. We visualise the training dynamics across iterations for two settings: maintaining separate weights (Left) versus sharing parameters (Right). (a) reports the evaluation performance, and (b) the training rewards. We present only the first iteration's reward curves for brevity.

We study the trade-off between parameter efficiency and training stability by comparing shared- and separate-weight designs (Figure 6 bottom and Table 1). Using *separate weights* yields stable optimisation, with the IDM score improving monotonically from 6.37 to 6.57 over three iterations (Figure 6, Left). In contrast, *shared weights* ($\theta = \phi$) reduce memory footprint but introduce instability, as reflected by a performance drop at Iteration 3 (to $\sim$6.52).

This instability is also reflected in downstream performance: on AURORA-BENCH, shared weights slightly underperform the separate variant (5.06 vs. 5; Table 1), and on long-horizon evaluation (Table 3), the shared model consistently lags behind (e.g., 1.74 vs. 1.68 overall). We attribute this gap to gradient interference between the heterogeneous objectives of FWM (visual generation) and IDM (inferring the linguistic action), suggesting that more robust unification mechanisms are needed to effectively share representations across modalities [41–44].

G IDM Analysis

AURORA-BENCH Evaluation. To validate the reliability of our reward mechanism, we evaluate the Inverse Dynamics Model (IDM) performance directly on AURORA-BENCH (Table 12). A fundamental premise of our framework is that inferring the action responsible for a state transition (IDM) is a more tractable task than generating high-dimensional future states (FWM). The results support this hypothesis: the IDM model achieves high action prediction accuracy (6.38 out of 10 as GPT score), serving as a strong discriminator. Additionally, we observe that the iterative paradigm can also leverage the reward signal from FWM, effectively improving GPT4o scores from 6.38 to 6.58 as our best approach. This stability is vital for our self-improving, ensuring that the reward signal remains informative and accurately penalises physical inconsistencies.

Linguistic Properties of Latent Actions. A potential concern in learning with the latent actions in SWIRL is reward hacking, where actions might degenerate into short, distinctive ciphers (e.g.,

single keywords or artifacts) to be hacked for an easier optimisable FWM. We analyse the evolution of complexities of the generated actions across three iterations for SWIRL in Figure 2b.

Contrary to the hypothesis of reward collapse, we observe that the uniqueness of generated actions remains consistently high ($> 94\%$) across all iterations for both shared and separate weight configurations. The similar observation happens for naturalness that the predicted actions are quite stably natural (as evidenced by the predicted perplexity of an LLM, GPT-2) throughout SWIRL's training iterations, without collapsing into any unnatural artifacts that could be easily hacked by FWM or IDM. Looking into the lengths of predicted actions, the model does not "shortcut" by generating brief tokens as well; instead, the average action length (around 16.8 tokens) consistently exceeds the ground truth average (around 7.2 tokens). This indicates that the IDM learns to provide more descriptive and detailed instructions to ensure the transition is consistently identifiable, rather than drifting into a simplified cipher. Qualitative inspection confirms that predictions remain physically meaningful and there is no model collapse happening on SWIRL (e.g., *"tearing paper into two pieces", "Swap the positions of the two objects.", "turning a bottle upside down"*), preserving natural language interpretability.

H Inference-time Verification

As a sanity check for SWIRL, we investigate the validity of the *coverage hypothesis* intrinsic to our GRPO formulation. For the IDM reward to effectively guide learning, the set of sampled rollouts must contain at least one candidate that sufficiently approximates the ground-truth future state. We evaluate the "Best-of-N" performance of Liquid-SFT, where the candidate with the highest GPT4o score is selected, by scaling the number of rollouts N. In Table 1, we observe a monotonic improvement for INFERENCE-TIME VERIFICATION as N increases, confirming that the base policy possesses the latent capability to generate high-fidelity transitions in multiple rollouts. This finding empirically validates our training premise: expanding the rollout space should increases the likelihood of capturing valid transitions, thereby providing the IDM discriminator with the necessary high-quality positive examples to reinforce.

I Detailed Results for Comparing SFT and SWIRL

Table 13 presents a detailed comparison between SWIRL and two supervised fine-tuning baselines under controlled data budgets. Across all five benchmarks and all training sample counts, RL consistently exhibits superior data efficiency compared to both *SFT-Continue* and *SFT-Merge*. While SFT baselines often plateau or even degrade as more samples are introduced, RL shows a clear trend of performance improvement as additional data becomes available.

Notably, RL begins to outperform SFT at relatively small data scales (6.4K samples) and the performance gap widens as training proceeds. On Action Genome, Something, and Whatsup, Kubric, which emphasises structured physical dynamics, RL achieves the strongest or near-strongest performance at every scale, indicating that reward-guided optimisation better captures latent world structure than direct imitation.

Aggregated across all benchmarks, RL achieves the highest average score at every data scale, with gains becoming increasingly pronounced at larger budgets ($4.27 \rightarrow 4.73$). These results suggest that RL not only improves final performance but also utilises limited data more effectively, aligning with our hypothesis that IDM's rewarding mechanism provides a stronger inductive bias for intrinsic world modeling than purely supervised objectives.

J Detailed Results for WORLDPREDICTIONBENCH

Table 14 presents a granular analysis of predictive performance across six consecutive autoregressive turns on *WorldPredictionBench*. This evaluation creates a challenging stress test for temporal consistency, as errors generated in early turns compound over the predictive horizon.

We observe two primary trends. First, as expected, all models exhibit performance decay as the horizon increases (N decreases as tasks are completed or fail). However, a clear divergence emerges between training paradigms. The direct supervised baseline (*Liquid-SFT*) suffers from rapid degrada-

Table 13: **Detailed Ablation: Data Efficiency (RL vs. SFT).** We report GPT-4o evaluation scores across five benchmarks and the aggregated average. We compare our proposed Reinforcement Learning (RL) fine-tuning against two Supervised Fine-Tuning (SFT) baselines: *SFT-Continue* (continual training with the additional samples) and *SFT-Merge* (we concatenate all samples and fine-tune model). Columns represent the number of training samples seen. **Bold** indicates the best performance for that specific sample count. Rows highlighted in blue denote our method.

		Number of Training Samples			
Benchmark	**Method**	**3.2k**	**6.4k**	**9.6k**	**12.8k**
MagicBrush	SFT-Continue	5.50	4.94	4.70	4.56
	SFT-Merge	**5.62**	5.56	4.80	5.62
	SWIRL (RL)	4.71	**5.56**	**6.61**	**6.19**
Action Genome	SFT-Continue	2.62	2.72	2.82	2.88
	SFT-Merge	**2.76**	2.98	2.94	2.58
	SWIRL (RL)	2.60	**3.32**	**3.67**	**3.38**
Something	SFT-Continue	2.90	2.86	2.70	2.74
	SFT-Merge	2.92	2.54	2.66	2.56
	SWIRL (RL)	**3.42**	**3.31**	**3.10**	**3.38**
Whatsup	SFT-Continue	3.18	3.69	3.16	3.51
	SFT-Merge	2.69	2.62	2.74	2.86
	SWIRL (RL)	**3.80**	**4.26**	**4.04**	**4.12**
Kubric	SFT-Continue	5.81	5.51	5.90	5.79
	SFT-Merge	6.68	6.66	**6.69**	6.49
	SWIRL (RL)	**6.90**	**6.70**	6.51	**6.65**
Average (All)	SFT-Continue	3.98	3.94	3.84	3.88
	SFT-Merge	4.15	4.07	3.95	4.01
	SWIRL (RL)	**4.27**	**4.63**	**4.67**	**4.73**

tion, dropping from an overall score of 3.09 in Turn 1 to 1.17 by Turn 4. In contrast, our proposed SWIRL demonstrate significantly improved robustness against this compounding error. The *Ours (Best)* configuration, maintains a score of 1.59 at Turn 4. This indicates that the reciprocal cycle does not merely memorise single-step transitions but internalises more robust physical dynamics that persist over long-horizon rollouts. While large-scale unified models like *Bagel* provide a high upper bound, our method close the gap compared to SFT baseline.

Table 14: **Long-Horizon Evaluation on WorldPredictionBench.** We report GPT-4o evaluation scores across 6 consecutive prediction turns. We compare our proposed method variants against the direct SFT baseline (Liquid-SFT), the strong unified baseline (Bagel), and other state-of-the-art VLMs. Rows highlighted in blue denote our methods. The sample count (N) decreases in later turns as completed tasks are filtered out. **Bold** indicates the best performance among the Liquid-based family (SFT vs. Ours).

Method	**COIN**	**CrossTask**	**EgoExo**	**EPIC**	**IKEA**	**Overall**
Turn 1 ($N = 400$)						
Bagel	4.71	4.46	3.55	3.42	4.27	4.29
Liquid-SFT	3.37	3.41	2.75	3.20	2.57	3.09
Ours	3.11	3.93	2.70	2.88	2.41	2.95
Ours (Best)	3.54	3.27	2.75	3.20	**2.61**	3.16
Other Baselines						

Continued on next page

Table 14 – continued from previous page

Method	COIN	CrossTask	EgoExo	EPIC	IKEA	Overall
OmniGen2	3.44	3.59	2.26	2.73	2.67	3.05
BLIP3o-NEXT	2.65	2.76	2.15	2.33	2.20	2.46
OmniGen-v1	3.41	3.46	2.64	4.35	2.73	3.25
UniWorld-V1	3.26	3.59	3.23	2.83	3.89	3.40
Turn 2 ($N = 400$)						
Bagel	4.37	4.34	3.75	3.38	4.04	4.10
Liquid-SFT	2.42	2.37	1.83	2.08	1.52	2.09
Ours	2.13	2.07	**2.28**	2.25	1.65	2.04
Ours (Best)	**2.66**	**2.90**	2.11	**2.78**	**1.79**	**2.42**
Other Baselines						
OmniGen2	3.13	2.80	2.09	2.38	2.08	2.64
BLIP3o-NEXT	2.05	1.80	1.58	1.65	1.60	1.82
OmniGen-v1	2.99	3.22	2.34	3.55	2.37	2.83
UniWorld-V1	3.25	3.49	3.06	3.38	3.86	3.41
Turn 3 ($N = 400$)						
Bagel	3.96	3.27	2.85	3.30	3.04	3.47
Liquid-SFT	1.46	1.27	1.42	**1.75**	1.17	1.40
Ours	1.65	1.51	1.75	1.25	1.31	1.53
Ours (Best)	**2.02**	**2.02**	**1.96**	1.73	**1.46**	**1.85**
Other Baselines						
OmniGen2	4.16	4.34	3.30	4.00	4.14	4.05
BLIP3o-NEXT	1.25	1.05	1.06	1.25	1.24	1.20
OmniGen-v1	4.05	3.95	3.81	2.93	4.48	4.00
UniWorld-V1	3.57	3.90	3.25	2.93	4.35	3.68
Turn 4 ($N = 210$)						
Bagel	4.21	3.70	2.53	3.88	2.64	3.22
Liquid-SFT	1.20	1.20	1.29	1.32	1.04	1.17
Ours	1.38	**1.60**	1.40	1.15	1.10	1.24
Ours (Best)	**2.20**	**1.60**	**1.43**	**1.71**	**1.32**	**1.59**
Other Baselines						
OmniGen2	4.40	3.40	3.29	3.18	3.95	3.75
BLIP3o-NEXT	1.00	0.70	1.00	1.06	0.89	0.95
OmniGen-v1	4.00	3.30	3.79	3.21	4.31	3.92
UniWorld-V1	3.00	2.70	3.83	2.44	4.32	3.59
Turn 5 ($N = 140$)						
Bagel	—	—	2.47	3.26	2.11	2.47
Liquid-SFT	—	—	1.23	1.30	0.91	1.07
Ours	—	—	**1.57**	**1.27**	1.05	1.23
Ours (Best)	—	—	1.54	1.07	**1.19**	**1.25**
Other Baselines						
OmniGen2	—	—	2.91	2.83	3.21	3.06
BLIP3o-NEXT	—	—	0.91	1.03	0.71	0.83
OmniGen-v1	—	—	3.57	2.57	3.43	3.28
UniWorld-V1	—	—	3.29	2.10	3.40	3.09

Continued on next page

Table 14 – continued from previous page

Method	COIN	CrossTask	EgoExo	EPIC	IKEA	Overall
Turn 6 ($N = 115$)						
Bagel	—	—	2.13	3.00	1.95	2.23
Liquid-SFT	—	—	1.04	1.17	0.88	0.97
Ours	—	—	1.23	1.17	**1.05**	**1.11**
Ours (Best)	—	—	**1.23**	1.22	0.97	1.08
Other Baselines						
OmniGen2	—	—	2.54	2.96	3.39	3.11
BLIP3o-NEXT	—	—	0.73	0.91	3.55	0.63
OmniGen-v1	—	—	3.04	2.22	4.05	3.45
UniWorld-V1	—	—	2.58	1.74	0.50	2.97

K Detailed Results for Iterative Evaluation

Table 15: **Detailed Iterative Dynamics (FWM & IDM).** We report the detailed breakdown of performance across three iterations for our two ablation settings: (1) Separate weights for Generator and Critic, and (2) Shared weights. Columns under "FWM Evaluation" represent the generation quality across five benchmarks. The "IDM Score" column represents the Critic's alignment accuracy. **Bold** numbers indicate the peak performance within each experimental setting.

		FWM						IDM
Setting	**Iteration**	**MagicBrush**	**AG**	**Something**	**Whatsup**	**Kubric**	**Avg**	**Avg Score**
Option 1: Separate Weights (No Sharing)								
	Iter 0	6.24	3.36	3.72	4.32	**7.14**	4.96	6.37
	Iter 1	**6.62**	**3.52**	**3.96**	**4.32**	6.96	**5.06**	6.52
	Iter 2	6.42	3.44	3.82	4.20	6.96	4.98	**6.56**
Option 2: Shared Weights ($\phi = \theta$)								
	Iter 0	6.24	3.36	3.72	4.32	7.14	4.96	6.37
	Iter 1	6.00	**3.38**	**3.73**	**4.46**	**7.45**	**5.00**	**6.57**
	Iter 2	**6.38**	3.18	**3.73**	4.32	7.27	4.97	6.52

Table 15 examines the iterative training dynamics of Forward World Modeling (FWM) and Inverse Dynamics Modeling (IDM) under two architectural choices: using separate weights for the FWM and IDM versus shared weights. We report results across three training iterations to highlight peak performance and learning progression.

The separate-weight configuration achieves the highest peak FWM performance, with Iteration 1 attaining the best average FWM score (5.06) and consistently strong results across all benchmarks. In particular, gains on MagicBrush, Action Genome, and Something indicate that decoupling the FWM and IDM allows each component to specialise more effectively, leading to higher-quality forward predictions when the system is optimally aligned.

While the shared-weight setup exhibits competitive and stable behaviour, it does not surpass the peak generative performance achieved by the separate-weight model. Notably, although IDM accuracy continues to improve slightly in later iterations for both settings, the separate-weight design reaches its optimal balance between FWM quality and IDM alignment earlier in training. This suggests that architectural decoupling enables more expressive forward modeling, even if later iterations yield diminishing returns.

Overall, these results indicate that separating FWM and IDM parameters is advantageous for maximising peak forward world modelling performance, motivating our choice of the separate-weight configuration in the main experiments where peak capability is the primary objective.

Table 16: **Ablation on GRPO Rollout Size (G).** We report the peak *Assemble* score achieved across all checkpoints for each rollout configuration. While larger group sizes ($G = 64$) achieve the best performance, moderate group sizes ($G = 16$) achieve comparable performance with much less compute.

#Rollout	MB	AG	Something	Whatsup	Kubric	Avg.
$G = 8$	6.04	3.38	3.45	4.26	6.98	4.80
$G = 16$	5.78	3.24	3.60	4.50	6.98	4.80
$G = 32$	5.72	3.30	3.37	4.18	6.80	4.68
$G = 64$	5.98	3.50	3.80	4.16	7.06	4.90

L Ablation on GRPO Rollout Size.

We analyse the effect of rollout size. The group size G is a critical hyperparameter in GRPO, as it governs the accuracy of the baseline estimation and the diversity of the exploration within each optimisation step. We empirically evaluate $G \in \{8, 16, 32, 64\}$ across AURORA-BENCH, with results summarised in Table 16. We observe that scaling the rollout size generally improves model performance, achieving a peak average score of 4.90 at $G = 64$. We attribute this to the reduced variance in advantage estimation: a larger pool of generations provides a more robust approximation of the expected return, enabling the policy to distinguish high-quality trajectories more effectively. Notably, while $G = 64$ yields the best absolute performance—particularly in complex environments like Kubric, the smaller configuration of $G = 16$ remains highly competitive (4.80 avg.) while requiring much less memory and compute during the generation phase.

M Qualitative Examples

We provide some qualitative examples in Figure 7 to illustrate the visual predictions of SWIRL across three distinct benchmarks: AURORA-BENCH, BYTEMORPH, and WORLDPREDICTIONBENCH.

AURORA-BENCH. The top panel demonstrates results on the AURORA benchmark, covering various subsets such as MagicBrush, Something, Emu, Kubric, and Whatsup. These examples highlight the model's capability to perform diverse action-centric image editing tasks, ranging from the general editing (e.g., Add a `supernova explosion in the sky`) and background replacement (e.g., `changing to Yellowstone National Park`) to precise geometric transformations and spatial reasoning. For instance, in the Something and Kubric subsets, the model successfully executes actions concerning physical regularities like `flip the bottle upside down` and distinct spatial rearrangements like `moving a specific plate to the left`.

BYTEMORPH. The middle panel displays qualitative results for BYTEMORPH, focusing on fine-grained control over camera and object dynamics. We visualise three distinct categories: Camera Zoom, Camera Motion, and Object Motion. The results show the model's ability to synthesise coherent view changes, such as zooming out to reveal context (e.g., *the coastline and a second boat*) or shifting the camera angle upward. Additionally, the Object Motion example demonstrates the generation of localised movement, specifically raising an animal's head while maintaining scene consistency.

WORLDPREDICTIONBENCH. The bottom panel illustrates long-horizon predictions on the WORLDPREDICTIONBENCH. Here, we show multi-step predictions where the model generates subsequent states based on the previous predicted image as the context and textual action descriptions. The examples depict long-horizon consistency in procedural tasks, such as replacing a car key battery (Start $\rightarrow$ Put in battery $\rightarrow$ Close cover) and arranging bedding (Start $\rightarrow$ Take out cover $\rightarrow$ Arrange nicely).

N Safeguards

We discuss the safeguards relevant to the responsible release of SWIRL, the models trained under our framework, and the data pipeline used during self-improvement.

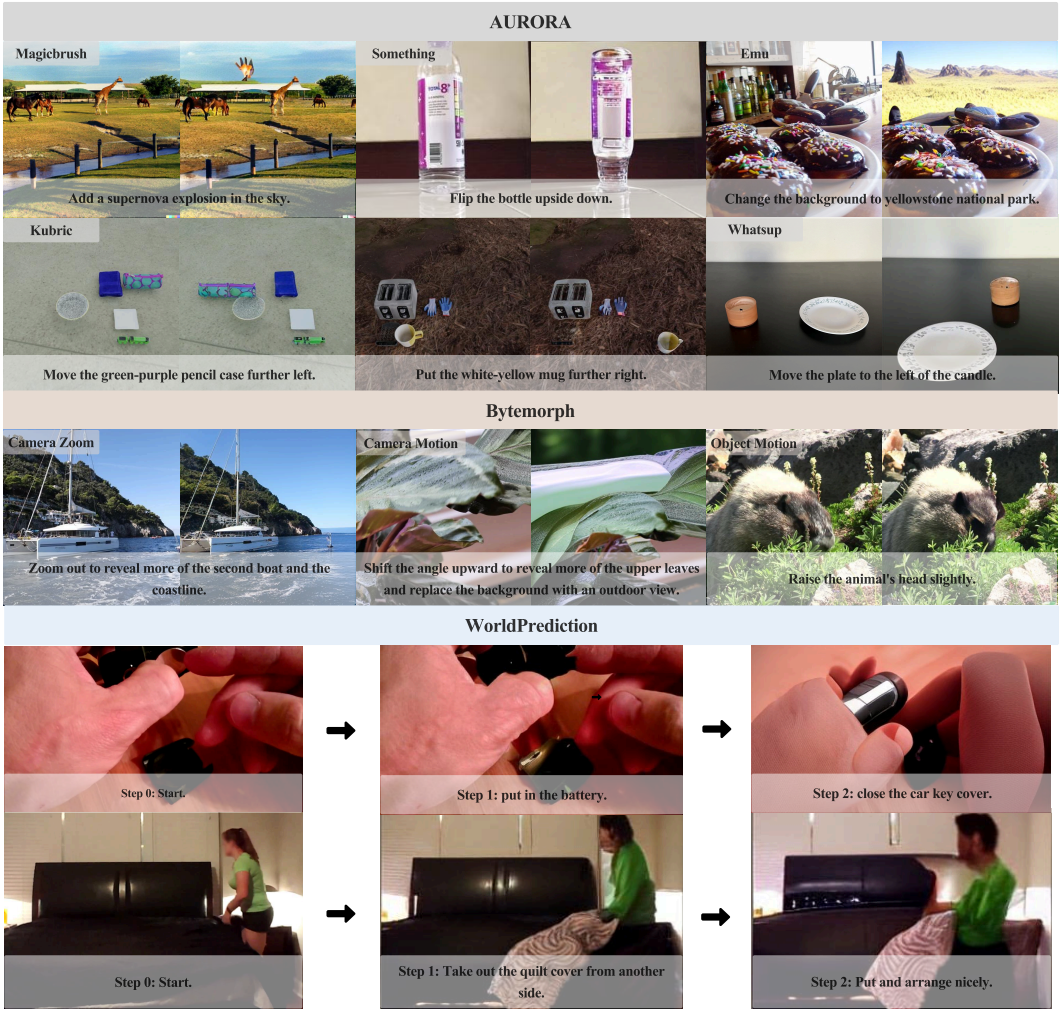


Figure 7: Qualitative examples produced by SWIRL for AURORA-BENCH, BYTEMORPH and WORLDPREDICTIONBENCH. We present the sample for each subset in these benchmarks.

Risk profile of released artefacts. The artefacts associated with this work are: (i) the SWIRL training code, (ii) the SFT-warmed and SWIRL-trained checkpoints of Liquid-7B for visual world modelling, and (iii) the SWIRL-trained checkpoints of Qwen-2.5-3B-Instruct for textual environments. None of these artefacts introduces a fundamentally new generative capability beyond what is already available in the underlying pre-trained backbones (Liquid [38] and Qwen-2.5 [47]), both of which are publicly released. SWIRL improves the *action-conditioned consistency* of forward predictions rather than unlocking novel content categories; the marginal misuse risk introduced by our checkpoints is therefore incremental rather than categorical. We nonetheless take the following concrete steps.

Training-data safeguards. The unlabelled video corpus used during the self-improvement loop (VIDGEN-1M [53], UCF-101 [50], MOMENTS-IN-TIME [51], KINETICS-700 [52]) consists of publicly released research datasets that have undergone curation by their original authors. For the SFT warm-up, we use PICO-BANANA-400K [48] and AURORA [49], both of which are released under research licences with their own content policies. We did not scrape additional web data, and we did not collect any new human-subject data. We performed a manual inspection of randomly sampled IDM outputs across iterations (Appendix G, Figure 2) and observed no degeneration into harmful, identifying, or otherwise unsafe action descriptions; latent actions remained natural-language, descriptive, and physically grounded (e.g., "tearing paper into two pieces", "swap the positions of the two objects"). This is consistent with our design: the IDM is rewarded by the FWM's likelihood

of reconstructing innocuous next frames drawn from research video corpora, which provides no incentive to generate harmful action labels.

Generative-misuse safeguards (visual checkpoints). The Liquid-based checkpoint can synthesise next-frame predictions conditioned on a source image and a textual action. The most plausible misuse vector is non-consensual image manipulation or low-quality deepfake generation. To mitigate this, we will (i) release the visual checkpoints under a *research-only* licence that explicitly prohibits use for impersonation, harassment, non-consensual intimate imagery, and political disinformation; (ii) require gated access via a request form that records intended use; (iii) ship the checkpoints with a model card that documents known failure modes and explicit prohibitions.

Agentic-misuse safeguards (web and tool-calling checkpoints). The Qwen-2.5-3B checkpoint trained on MIND2WEB and STABLETOOLBENCH predicts the *outcome* of web interactions and API calls; it does not itself execute actions in the real world. Even so, an improved internal model of web/tool dynamics could in principle assist an attacker in planning automated interactions (e.g., bypassing CAPTCHAs, scaling scraping). We mitigate this by (i) training only on publicly released benchmark trajectories from MIND2WEB [56] and STABLETOOLBENCH [57], both of which were collected under research protocols and contain no privileged endpoints; (ii) not releasing any execution harness; and (iii) releasing the checkpoint under the same gated, research-only licence described above.

Operational recommendations. We recommend that any deployed system built on SWIRL checkpoints incorporate (i) input/output content filters, (ii) watermarking or provenance metadata for synthesised images (e.g., C2PA), (iii) rate limits and audit logging for autonomous web agents, and (iv) human-in-the-loop review for any action sequence executed in a real environment. Future work extending SWIRL should additionally consider constitutional-AI-style reward shaping or safety classifiers within the IDM reward signal, as flagged in our Impact Statement.

We believe these safeguards are commensurate with the marginal capability uplift introduced by this work, while preserving the scientific value of releasing reproducible artefacts.

O Declaration of LLM Usage

We declare that the large language model (LLM) was only used to assist in minor tasks, including revising the manuscript for grammatical correctness, improving phrasing, and performing small technical implementations such as debugging code snippets. All scientific ideas, results, analyses, and conclusions presented in this paper are entirely the work of the authors.

P Licences

In this section, we list the public assests we used in this paper and the corresponding links.

Datasets. We include the detailed license and URL for the datasets we used in this paper.

- AURORA and AURORA-BENCH [49]: MIT license, the reader can find the corresponding version we use in this paper in `https://github.com/McGill-NLP/AURORA`.
- Movements-in-Time [51]: BSD-2-Clause license and its own License for Non-Commercial Use, the reader can find the corresponding version we use in this paper in `http://moments.csail.mit.edu/`.
- UCF-101 [50]: unknown license, the reader can find the corresponding version we use in this paper in `https://huggingface.co/datasets/flwrlabs/ucf101`.
- Kinetics700 [52, 64]: Creative Commons Attribution 4.0 International License, the reader can find the corresponding version we use in this paper in `https://research.google/pubs/the-kinetics-human-action-video-dataset/`.
- VidGen-1M [53]: Apache-2.0 license, reader can find the corresponding version we use in this paper in `https://huggingface.co/datasets/Fudan-FUXI/VIDGEN-1M`.
- ByteMorph [63]: Creative Commons Zero v1.0 Universal license, reader can find the corresponding version we use in this paper in `https://huggingface.co/datasets/Boese0601/ByteMorph-Bench`.

- WorldPredictionBench [19]: Attribution-NonCommercial 4.0 International license, reader can find the corresponding version we use in this paper in `https://github.com/facebookresearch/WorldPrediction`.
- StableToolBench [57]: Apache-2.0 license, reader can find the corresponding version we use in this paper in `https://github.com/THUNLP-MT/StableToolBench`.
- ScienceWorld [55]: Apache-2.0 license, reader can find the corresponding version we use in this paper in `https://github.com/allenai/scienceworld`.
- Mind2Web [56]: MIT license, reader can find the corresponding version we use in this paper in `https://github.com/OSU-NLP-Group/Mind2Web`.

Model. We use the other following checkpoints for the implementation:

- Liquid [38]: MIT license, reader can find the corresponding version we use in this paper in `https://huggingface.co/Junfeng5/Liquid_V1_7B`.
- Qwen-2.5-Instruct [56]: `qwen-research` license, reader can find the corresponding version we use in this paper in `https://huggingface.co/Qwen/Qwen2.5-3B-Instruct`.

Implementation. We use the other following code for the implementation:

- Transformers [65]: Apache-2.0 license. We use the 4.38.0 version, following the link at `https://github.com/huggingface/transformers`.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Abstract and introduction include the claims made in the paper.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We have a section for Limitations.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: Discussed in Appendix A.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Discuss the hyperparameters and reproducibility in Appendix D. We will also release the code if the paper is accepted.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: As the footnote in Abstract, we will release the code if the paper gets accepted.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: The details are presented in Appendix D.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We conduct and provide the sensitivity analysis in Appendix C.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.

- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: We provide the compute budget in Appendix D.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: We have read and review the NeurIPS Code of Ethics.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: We have a section dedicated for Boarder Impacts.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [Yes]

Justification: Provided in Appendix N.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: Check the assets and licenses in Appendix P.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: We properly provide the documentation for the releasing code and our checkpoints used in this paper, together with the necessary license.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: No crowdsourcing and human subjects have involved in this paper.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: No crowdsourcing and human subjects have involved in this paper.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: Discussed in Appendix O.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.