World-in-Loop: Online Correction via Event-Triggered World Models for Robust VLA Policies

Shaoqing Xu[1,2,*], Fang Li[1,2,*], Yuechen Luo[3], Qimao Chen[3], Yifan Yang[2], Zhixiang Duan[2†], Long Chen[2†], and Zhi-xin Yang[1✉]

[1] The State Key Laboratory of Internet of Things for Smart City, Centre for Artificial Intelligence and Robotics, and Department of Electromechanical Engineering, University of Macau, Macau SAR
[2] Xiaomi Robotics and Autonomous Driving, Beijing, China
[3] Tsinghua University, Beijing, China
{shaoqing.xu, Fang.lee}@connect.um.edu.mo

Abstract. Large-scale Vision-Language-Action (VLA) models excel at mapping natural language instructions to robotic actions. However, a primary bottleneck in generalizing these models is the scarcity of high-quality real-world data. While fine-tuning requires massive successful demonstrations, the low success rates of current base policies in dynamic environments often lead to compounding errors and inefficient, low-quality data collection. To enhance both real-world data acquisition and long-horizon task success, we propose a proactive, online intervention framework. Our core innovation is a lightweight, composite World Model (WM) acting as an event-triggered "corrector". Instead of continuous frame-by-frame rollouts across the entire task horizon, our WM intervenes only at critical decision points (e.g., prior to grasping) to evaluate the proposed action's feasibility. If a high failure risk is detected, the system proactively rolls back to a heal state. A generative model then synthesizes a short video of a successful future trajectory, which guides the robot's re-attempt via an action decoder. We validate our framework across multiple robotic grasping tasks on both simulation and real-world systems. Extensive results demonstrate that deploying world models as active, on-demand correctors significantly improves task success rates and efficiently harvests high-quality execution data by rescuing potential failures before they occur.

1 Introduction

Vision-Language-Action (VLA) models [4, 16, 19, 20, 26, 28, 33, 39, 41, 44] have emerged as a powerful paradigm for robotic manipulation, enabling agents to interpret natural language instructions and generate corresponding actions from

* Equal contribution. † Project leader. ✉ Corresponding author.

visual input. Despite their success, these systems rely heavily on imitation learning derived strictly from ***positive grasp data***. Consequently, they typically treat action generation as a terminal, open-loop step without mechanisms for dynamic supervision which often results in execution bias and compounding errors, particularly in dynamic or ambiguous environments. Therefore, it often happens that remedial measures are taken after the sub-task fails. Crucially, in real-world robotic deployment, waiting to correct these errors until after a physical failure has occurred is often too late. Such low success rates not only hinder reliable deployment but also exacerbate a primary bottleneck in robotic generalization: the scarcity of high-quality real-world data. While current policies frequently fail in operation, data collection is highly inefficient and often yields suboptimal trajectories. To overcome this data bottleneck, there is a pressing need for a proactive, online remedial mechanism that must be capable of predicting failure risks at critical moments to rectify flawed actions before the actual failure materializes in time, thereby simultaneously improving long-horizon task success and data collection efficiency.

Parallel to the development of VLAs, world models (WMs) [2, 8, 36] have gained significant attention for their ability to model environmental dynamics and support predictive reasoning which enable agents to anticipate consequences and plan accordingly. However, prior methods typically mandate continuous future simulation across the entire task horizon, these exhaustive, long-sequence rollouts incur prohibitive computational costs, making them impractical for real-time control. Their training is also commonly restricted to successful demonstrations, limiting their ability to generalize from or recover from failure scenarios.

To address the limitations of both paradigms, we ask: ***what if an agent could pause at critical moments, assess the feasibility of its own plan, and imagine a trajectory to success before committing to a potentially flawed action?*** In this work, we propose world-in-loop, a framework for online policy rectification, which synergistically incorporates the generalized capabilities of VLAs with the predictive power of a lightweight, event-driven world model. Our approach is designed to address two key limitations in current robotic manipulation pipelines: (1) the inability to robustly detect and respond to grasp failures in real time, and (2) the underutilization of world models in guiding corrective actions. Specifically, our method begins with a base VLA model controlling the robot's actions until it outputs a grasp command, which designates the current visual frame as a keyframe suitable for grasping. At this juncture, our framework first leverages a finetuned discriminative model to evaluate the risk of failure for the proposed action, which is trained on the data composed of ***high-value failure cases***. If a potential failure is detected, a generative world model "imagines" a short video of a successful grasp from the current state. This imagined success trajectory is then decoded to actions executed by robot, which guide the robot into the correct state.

This “in-the-loop" architecture is designed for synergy and efficiency. By activating the WM only when necessary, it avoids the computational overhead of

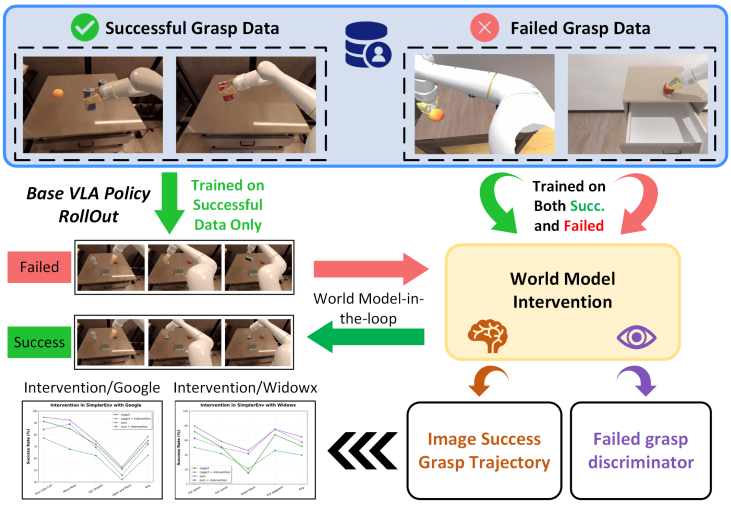


Fig. 1: An overview of our core idea. Standard VLAs (left) suffer from compounding errors, leading to task failure, which is trained only on successfully collected data. Our World-in-Loop framework (right) intervenes at critical moments, evaluating the proposed action and leveraging a world model to imagine a successful outcome by incorporating both successful and failed collected data, which then guides a corrected action to ensure task success.

continuous prediction. Crucially, by learning from potential failures, it enhances data efficiency and robustness. In summary, our contributions are as follows:

- We propose a novel and efficient integration of Vision-Language-Action (VLA) models with world models (WM), where the WM operates as a lightweight, event-driven plug-in that is selectively activated only when a grasp action is predicted to break the efficient bottleneck. This design enables real-time correction without incurring the cost of continuous rollouts.
- Our framework is the first work to apply world models for grasp correction, leveraging their discriminative ability to assess graspability and generative capacity to predict a single future graspable state—thus avoiding the continuous resource-intensive generation used in prior methods.
- We introduce a failure-driven training strategy for the world model, utilizing both successful and failed executions to learn robust decision boundaries. This improves data efficiency and generalization, leading to state-of-the-art performance across multiple robotic grasping benchmarks with strong robustness to online intervention and environmental disturbances.

2 Related Works

Vision-Language-Action (VLA) Models. The recent success of Large Language Models (LLMs) [1,3]and Vision-Language Models (VLMs) [17,43] has led

to the development of Vision-Language-Action (VLA) models [6, 11, 18, 23, 24], which enable robots to interpret multi-modal inputs and generate context-aware actions. Early pioneers like VIMA [13] demonstrated the potential of a single, massive network to handle diverse tasks, followed by more specialized robotics models like RT-1 [4], and RT-2 [44], which showed that co-fining vision-language models on robotics data could transfer abstract web-scale knowledge to physical control, enabling impressive zero-shot generalization. More recent efforts [19,29,40], have integrated more powerful vision encoders, further enhancing the models' perceptual understanding. However, these models typically rely on imitation learning and treat action generation as a terminal output, which limits their adaptability in dynamic environments. The problem of online error accumulation in unseen situations remains a significant open challenge. Our work directly addresses this limitation not by changing the VLA architecture itself, but by augmenting it with an external, intelligent correction loop.

World Models in Robotics. World Models, which learn a predictive model for environment, enabling agents to simulate future states and plan accordingly from current observations and actions. Seminal works like PlaNet [10] and Dreamer series [9] have shown that agents can learn complex behaviors by "dreaming" or planning within a learned latent space. WorldVLA [5] attempted to unify VLA and world models into a single framework, highlighting the mutual benefits of joint training. Furthermore, recent World Action Models(WAM) [15, 27, 38, 42] have further explored the value of learning potential action priors from the video world model. However, these models typically require continuous, step-by-step prediction of future states (often as images or latent vectors) for planning. They served merely as an auxiliary head to provide an extra supervision signal during training, rather than actively guiding the online correction of powerful learning policy, lacking explicit mechanisms for failure recovery. Our key distinction is the repurposing of WM components (generation and discrimination) for on-demand, event-triggered intervention, avoiding the overhead of constant future simulation.

Online Policy Correction and Intervention. The concept of online correction is crucial for deploying robots in real-world environments. Traditional grasping pipelines [7,14] often rely on static policies or human intervention when failures occur, limiting their adaptability in ambiguous environments. A prominent direction leverages the reasoning capabilities of Large Vision-Language (LVLMs) [25, 30, 31] to enable robots to reflect on failed grasp attempts and adjust strategies accordingly. Other approaches, such as Phoenix [37] proposes a motion-based self-reflection framework that translates high-level semantic feedback into fine-grained action corrections, enabling precise recovery from manipulation errors. However, these methods either require human oversight or struggle to generalize to the vast number of potential failure modes. In contrast, our framework introduces a proactive, perceptually-grounded self-correction loop that leverages the predictive power of a world model. By combining discriminative evaluation and generative prediction, our system can assess graspability and simulate future graspable states in a lightweight and targeted manner.

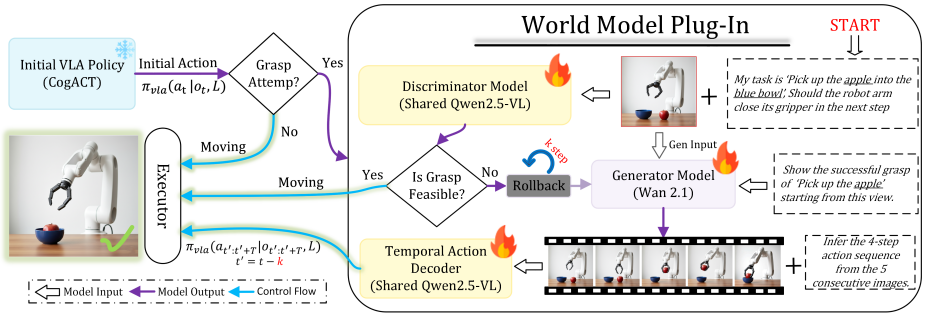


Fig. 2: The architecture of our ***World-in-Loop*** framework. The process follows a 'Propose-Evaluate-Imagine-Correct' loop. When a keyframe (grasp initiation) is detected, a Discriminator evaluates the proposed action. If failure is predicted, a Generator generates a video of successful grasp. This video is then fed back to the VLA, which outputs a corrected, final action. Note the weight-sharing between the Discriminator and the corrective VLA module for efficiency.

3 Methodology

Our proposed method, ***World-in-Loop***, repurposes the concept of a World Model (WM) as a novel online intervention mechanism to correct base VLA policy at critical decision points. Unlike in traditional applications where WMs continuously predict future states, our approach utilizes a lightweight, composite WM as an on-demand "corrector" that combines discriminative and generative capabilities. This section first formulates the problem of online correction, then details our World-in-Loop framework, and finally elaborates on the specific design of our composite world model components.

Problem Formulation. We formulate the robotic manipulation task as a Partially Observable Markov Decision Process (POMDP). Let the observation space be $\mathcal{O}$, at each timestep t, the agent receives a visual observation $o_t = (I_t, s_t) \in \mathcal{O}$ combines an RGB image $I_t \in \mathcal{I}$ and proprioceptive state $s_t \in \mathcal{S}$ (e.g., end-effector pose, gripper state). A pre-trained VLA policy $\pi_{\text{VLA}}(a_t|o_t, L)$, trained via behavioral cloning on an expert dataset $\mathcal{D}_{\text{expert}} = \{(o_i, a_i)\}_{i=1}^{N}$, predicts low-level action $a_t = (\Delta xyz_t, \Delta rpy_t, g_t)$, where $(\Delta xyz_t, \Delta rpy_t)$ denotes Cartesian and orientation increments, and g_t is the gripper command (open/close) which is conditioned on the observation and a language instruction L. The training objective minimizes the deviation from expert actions:

$$\pi_{\text{VLA}} = \arg\min_{\pi} \sum_{(o,a)\in\mathcal{D}_{\text{expert}}} \mathcal{L}_{\text{imitation}}(\pi(o, L), a) \tag{1}$$

The central challenge during deployment is the covariate shift, where the online observation distribution $P_{\text{online}}(o)$ deviates from the training distribution $P_{\text{expert}}(o)$. This causes the performance of π_{VLA} to degrade, leading to an expected return $\mathbb{E}_{\tau\sim\pi_{\text{VLA}}}[\sum_t R(s_t, a_t)]$ that falls below an acceptable threshold.

To solve this, we introduce an online correction function $\mathcal{C}$, which modulates the VLA's proposal $a_t = \pi_{\text{VLA}}(o_t, I)$ to produce a final action $a'_t = \mathcal{C}(o_t, a_t)$. We instantiate our World Model as $\mathcal{C}$, which intelligently intervenes to guide the policy back toward a successful trajectory and maximizes the expected return of the resultant policy π' while minimizing computational overhead by invoking $\mathcal{C}$ sparingly, denoted as: $\max_{\mathcal{C}} \mathbb{E}_{\tau\sim\pi'}[\sum_{t=0}^{T} \mathcal{R}(s_t, a'_t)]$

3.1 Base Policy: Keyframe Detection via Primary VLA

The first step in our framework is to leverage a pre-trained VLA model, e.g., CogACT, as the base policy to generate an initial expert's action, a_t. For manipulation tasks like grasping, the most decisive moment is the initiation of the grasp itself. Therefore, we define the keyframe trigger as the time where the base VLA policy proposes an action with gripper state transitioning to "close". The supervisor takes control and first evaluates the proposed action's likelihood of success using a discriminative module, D_t, at this keyframe. For all other proposed actions, such as moving the arm or opening the gripper, the supervisor remains inactive, and the base policy is executed directly without intervention.

3.2 Intervener: A Composite World Model

Our WM is not a monolithic dynamics predictor but a composite system of two specialized components: a VLM-based discriminator for assessing feasibility and a video-generative model for imagining corrections.

The Discriminator. Is This Grasp Feasible? The discriminative module D_t is realized using a powerful Vision-Language Model (VLM), specifically a fine-tuned Qwen-VL 2.5 [3]. We formulate the failure prediction task as a Visual Question Answering (VQA) problem instead of training a specialized binary classifier for better generalization. The module receives the current observation o_t and a structured text prompt: "*Query: My task is 'Stack Green Block on Yellow Block', Should the robot arm close its gripper in the next step?*" The model, denoted as $\mathcal{M}_{\text{disc}}$, is trained on a curated dataset of both successful and failed grasp attempts, learning to output $D_t \in \{\text{Suitable, Unsuitabel}\}$. This process can be formally expressed as maximizing the probability of the output text given the visual and text inputs:

$$D_t = \text{argmax} P(d|o_t, c_{\text{disc}}; \theta_{\mathcal{M}_{\text{disc}}}), d \in \{\text{suitable, unsuitable}\} \tag{2}$$

This VQA formulation allows the VLM to leverage its rich, pre-trained understanding of object properties, spatial relationships, and physical affordances to make an informed judgment about the feasibility of the proposed grasp.

The Generator. When the discriminative component signals a potential failure (d_t = unsuitable), as shown in top row in Figure 3, the system performs a state rollback of k steps to retrieve a more stable, recent observation, $o_{t'}$, where $t' = t - k$, and then activate the *Generative Model*, $\mathcal{M}_{\text{gen}}$, such as a image-to-video diffusion model (e.g., WAN [35]). The generator receives the observation $o_{t'}$ and a

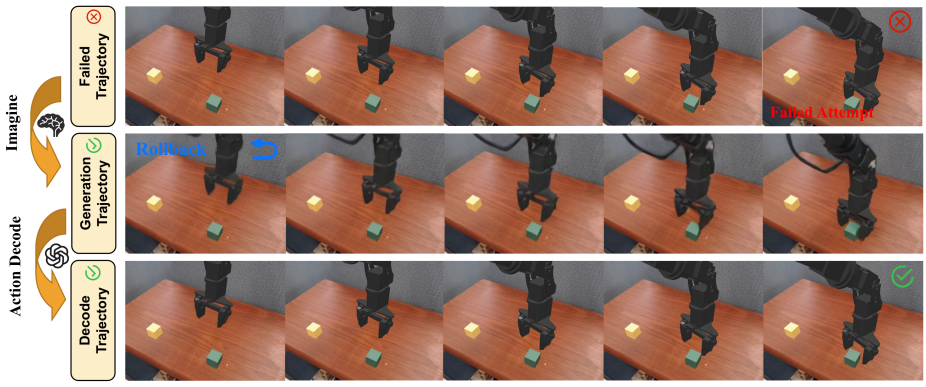


Fig. 3: The core correction mechanism of our framework. The top row shows a failed trajectory from the base VLA policy. Upon predicting failure, our system generates a successful trajectory (middle row). This imagined plan is then decoded and executed, leading to task success (bottom row).

high-level goal c_{gen}, e.g., "Show the successful grasp of 'Pick up the apple' starting from this view." It then synthesizes a short, physically plausible video sequence, $V_{\text{succ}} = (o_{t'}, o_{t'+1}, \ldots, o_{t'+T})$ that depicts a successful trajectory. This generation process is a conditional sampling from the model's learned distribution:

$$V_{\text{succ}} \sim P(V | o_{t':t'+T}, c_{\text{gen}}; \theta_{\mathcal{M}_{\text{gen}}}) \tag{3}$$

3.3 Temporal Action Decoder for Policy Refinement

The final and most crucial step of our framework is closing the corrective loop by feeding the generated knowledge back into the policy. We take the imagined successful video V_{succ}, presented in the middle row in Figure 3, to serve as a new, potent visual context for getting the successful trajectory. To maximize resource efficiency, we architect the discriminative module ($\mathcal{M}_{\text{disc}}$), which also serves as our behavior parser, acting as the Inverse Dynamic Model (IDM) by sharing its weights, which can generate refined action with a significantly higher likelihood of success, denoted as the bottom row in Figure 3. We enable this dual-role capability through a multi-task Question-Answering (QA) training scheme, where the model is trained on different QA formats corresponding to its roles as an evaluator and an actor, *e.g.* "*Query* : Infer the 4-step action sequence from the 5 consecutive images, output actions in order. Output format requirement:

- 1. The result must be a JSON array with exactly 4 elements.
- 2. Each element must be an array of 7 floating-point numbers.
- 3. Do not add explanations, text, or formatting outside the JSON array.

Answer : [[0.001, -0.002, -0.005, -0.016, -0.003, -0.013, 1.0], [0.001, 0.0, -0.013, -0.035, -0.011, -0.03, 1.0], [0.001, -0.003, -0.011, -0.032, -0.002, -0.006, 1.0], [0.001, -0.0, -0.006, -0.017, -0.002, -0.0, 0.0]]."

3.4 Training Objectives

Our framework is optimized holistically through a composite training objective designed to supervise each part of the rectification process. The base VLA policy doesn't need fine-tuning. Concurrently, the training of the world model is completely following WAN [35]. The discriminator and action decoder share the same VLM and are supervised with cross-entropy (CE) loss.

4 Experiments

4.1 Experimental Setup

Experimental Platforms. We evaluated our framework across both real-world and simulated environments. For real-world validation, we used four challenging long-horizon tasks on two distinct dual-arm platforms: the Xiaomi Robot, a wheeled system with two 7-DoF arms, and ALOHA, featuring two 6-DoF arms with parallel grippers. All real-world experiments were performed on a single NVIDIA RTX 3090 GPU, with 32 trials conducted per task. For simulation, we conducted experiments on two platforms, SIMPLER [21] and LIBERO [22], utilizing three types of robotic arms: WidowX, Google Robot, and Franka.
Initial Base VLA policy. To demonstrate the generalization ability of our framework, we conduct the experiment on various VLA policies, including CogACT [19], SpatialVLA [28] and MemoryVLA [29]. Crucially, we deliberately choose not to fine-tune initial policy, since fine-tuning actor policies typically requires a substantial volume of successful demonstration data, which is difficult and costly to collect, particularly on real-world robotic platforms.
World Model Training. Our composite world model consists of a discriminator and a generator. For the discriminator, we curated dataset by manually and semi-automatically labeling critical keyframes from demonstrations as either 'suitable' or 'unsuitable' for action transition. This dataset comprises 100k frames from subset of BridgeV2 [34] and Fractal and 2k frames collected from our real-world setups, with a balanced 1:1 ratio of positive and negative samples. An "unsuitable" state is defined as one in which the gripper's action transfer is not intended or physically feasible. Upon detecting such state, the robot arm rolls back m steps (default m=15) to acquire new observation. For the generation model, we use 33k video clips from the BridgeV2 and Fractal dataset. Additionally, for real-world fine-tuning, we collect 200 successful grasping demonstrations, 100 each from the Xiaomi Robot and the ALOHA system with each video comprising 5 to 20 frames.
Temporal Action Decoder. As noted in Sec. 3.3, the action decoder is shared with the discriminator. Specifically, we constructed a custom VQA dataset featuring imagined successful videos V_{succ} to guide the model in generating corrected action sequences, with ground truth labels derived from the corresponding source data. To ensure training efficiency, we utilize a multi-task QA structure that enables simultaneous optimization of both the decoder and the discriminator. For each training step, 5 frames are randomly selected around the key grasp-time.

The model takes an observation sequence $O_{t':t'+T}$ and a text prompt L as input and outputs an action sequence $a_{t':t'+T}$, with a default action horizon of T=5.

4.2 Evaluation Performance

To comprehensively validate the effectiveness, efficiency, and real-world applicability of the World-in-Loop, we conducted a series of extensive experiments in both standardized simulation environments and on a physical robot platform. The visualization of processing is also shown in Figure 5.

As shown in Table 1, our proposed online correction pipeline consistently improves performance across both simulation platforms (WidowX and Google Robot) and diverse base VLA policies, demonstrating strong generalizability.

- On SimplerEnv with WidowX. Our pipeline delivers larger improvements on grasping success and task success over various baselines (e.g., from 71.9% $\rightarrow$ 78.2% for MemoryVLA). Furthermore, for lower-performance VLA policy, CogACT, the improvement in success rate was even more significant, increasing by approximately 26.4%.
- On SimplerEnv with Google Robot. Our pipeline boosts success rates by +0.7%–5.7% over various baselines (e.g., from 61.3% $\rightarrow$ 67.0% for CogACT on Visual Matching) and significantly enhances Visual Matching and Variant Aggregation subtasks.

The improvement of multiple metics indicates that our pipeline can achieve effective generalization under different policies, providing a reliable correction strategy for real-world applications. Crucially, our framework can ***plug into any*** existing VLA policies or Video Action Models(mimic-video [27], Cosmos policy [15] etc.), without modifying the architecture of base policy. These results confirm that our work is not task-specific or platform-specific, but a general-purpose enhancement that reliably lifts performance across heterogeneous robotic setups and learning paradigms.

Real-World Set Up. The benefits of our correction framework are even more pronounced in the physical world, where unmodeled dynamics and visual noise are prevalent. Our method not only improves the grasp itself but also positively impacts the overall task success, indicating a more robust trajectory following the correction. As summarized in Table 2, we evaluated our framework on a series of challenging real-world tasks and compared it against the baseline VLA policy. The performance gap is particularly significant in tasks requiring high precision under uncertainty, such as "Grasp Watercup" (a 16.7% improvement in success) and "Grasp Block into Plate" (a 17.3% improvement). The difference between the "Grasp" and "Success" metrics is also revealing: even when the baseline manages a grasp, it may be unstable, leading to subsequent failure. Our method's higher final success rate indicates that the imagined correction provides a better-quality grasp, setting the robot up for a more stable and successful completion of the entire task.

Table 1: Comparison on SIMPLER with WidowX and Google Robot. † denote the method fine-tuned in domain. WiL is our World in Loop. Best results are in bold.

Method	SIMPLER with WidowX					SIMPLER with Google Robot									
	Tasks (Grasp/Success %)				Avg. (G/S)	Visual Matching (Success %)					Variant Aggregation (Success %)				
	Spoon on towel	Carrot on plate	Block on block	Put egg. on basket		Pick Coke	Move Near	O/C Drawer	Top Dr. Apple	Avg.	Pick Coke	Move Near	O/C Drawer	Top Dr. Apple	Avg.
RT-1†	—	—	—	—	—	85.7	44.2	73.0	6.5	52.4	89.8	50.0	32.3	2.6	43.7
RT-1-X	16.7/0.0	20.8/4.2	8.3/0.0	0.0/0.0	11.5/1.1	56.7	31.7	59.7	21.3	42.4	49.0	32.3	29.4	10.1	30.2
RT-2-X	—	—	—	—	—	78.7	77.9	25.0	3.7	46.3	82.3	79.2	35.3	20.6	54.4
Octo-Base	34.7/12.5	52.8/8.3	31.9/8.3	66.7/43.1	46.5/16.0	17.0	4.2	22.7	0.0	11.0	0.6	3.1	1.1	0.0	1.2
Octo-Small	77.8/47.2	27.8/9.7	40.3/4.2	87.5/56.9	58.4/29.5	—	—	—	—	—	—	—	—	—	—
OpenVLA	4.1/0.0	33.3/0.0	12.5/0.0	8.3/4.1	14.6/1.0	18.0	56.3	63.0	0.0	34.3	60.8	67.7	28.8	0.0	39.3
RoboVLMs†	70.8/45.8	33.3/20.8	54.2/4.2	91.7/79.2	62.5/37.5	77.3	61.7	43.5	24.1	41.9	—	—	—	—	—
SpatialVLA	25.0/20.8	41.7/20.8	58.3/25.0	79.2/70.8	51.2/34.4	81.0	69.6	59.3	—	71.9	89.5	72.7	41.8	—	**68.8**
ThinkAct†	—/58.3	—/37.5	—/8.7	–/70.8	—/43.8	92.0	72.4	50.0	—	—	84.0	63.8	47.6	—	—
SpatialVLA†	20.8/16.7	29.2/25.0	62.5/29.2	100.0/**100.0**	53.1/42.7	85.2	78.3	56.1	8.3	57.0	87.5	72.1	41.0	12.5	53.3
+WiL	**33.3/27.5**	**62.5/33.3**	**79.2/50.0**	**100.0**/95.8	**68.8/51.7**	**87.2**	**79.2**	**58.1**	**16.7**	**60.4**	**89.1**	**74.3**	**42.0**	**24.0**	**57.4**
CogACT	79.2/66.7	50.0/50.0	41.6/20.8	75.0/66.7	61.5/51.1	91.3	85.0	71.8	50.9	74.8	89.6	80.8	28.3	46.6	61.3
+WiL	**83.3/75.0**	**62.5/62.5**	**62.5/37.5**	**87.5/83.3**	**74.0/64.6**	**94.7**	**92.5**	**74.5**	**52.6**	**78.5**	**91.8**	**86.3**	**38.5**	**51.2**	**67.0**
MemoryVLA	79.2/75.0	79.2/70.8	54.2/41.7	**100.0/100.0**	78.2/71.9	90.7	88.0	**84.7**	47.2	77.7	80.5	78.8	**53.2**	58.3	67.7
+WiL	**87.5/79.2**	**83.3/79.2**	**70.8/54.2**	**100.0/100.0**	**85.4/78.2**	**93.1**	**89.2**	83.9	**48.1**	**78.6**	**82.6**	**79.3**	53.0	**59.8**	**68.5**

Table 2: Real-world results.

Setting	Grasp Watercup	Grasp Block into Plate	Move Block Near	Stack Block	Avg.
	Grasp (%).				
Baseline	66.7	75.0	75.0	66.7	70.9
Ours	79.2	91.7	87.5	83.3	**85.4**
	Success (%).				
Baseline	54.2	58.3	75.0	66.7	63.6
Ours	70.9	75.0	83.3	75.0	**76.1**

Table 3: Impact of rollback depth.

Setting	Put spoon on towel	Put carrot on plate	Stack green on yellow	Put eggplant on yellow	Avg.
	Success (%).				
Baseline	66.7	50.0	20.8	66.7	51.1
w/o Rollback	50.0	50.0	**41.7**	75.0	54.2
	Correction with Rollback				
Rollback 5 × *steps*	58.3	50.0	37.5	75.0	55.2
Rollback 10 × *steps*	66.7	58.3	**41.7**	79.2	61.5
Rollback 15 × *steps*	**75.0**	**62.5**	37.5	**83.3**	**64.6**
Rollback 20 × *steps*	66.7	58.3	**41.7**	75.0	60.4
Rollback 25 × *steps*	62.5	54.2	**41.7**	75.0	58.4

4.3 Ablation Study

Ablation on rollback mechanism. The results, presented in Table 3, systematically evaluate the impact of generative correction and the state rollback mechanism. First, we evaluate the base VLA policy as the performance floor. In the Correction w/o Rollback setting, the WM attempts to correct the policy from the immediate state where failure is predicted. We observe that this approach provides marginal benefits which suggests that by the time a failure is imminent, the state may already be irrecoverable, limiting the effectiveness of any correction attempted from that point. Second, we analyze the effect of Correction with Rollback at varying depths (5, 10, 15, 20 and 25 steps). The results show a clear and positive trend: increasing the rollback depth generally improves task success rates. By reverting to an earlier, "healthier" state, the generative model is provided a more viable starting point to imagine a successful trajectory. When rollback to 20 steps, video generation models suffer from "hallucinations" and temporal drift over long horizons introduces significant accumulated errors that the temporal action decoder cannot effectively resolve. This confirms that the state rollback is a critical component of our framework, enabling the agent to effectively escape near-failure states and replan a successful course of action.

Table 4: Ablation study on maximum intervention. We analyze how different retry constraints affect trigger frequency, success rate, execution steps, and overall efficiency.

Intervention Limit (Max)	Trigger Freq. (Times/Ep.)	Success Rate (%)	Avg. Exec. Steps	Avg. Latency (s/Ep.)	Efficiency (Ep./min)
Baseline (CogACT)	–	51.1	67.8	15.7	1.92
$Max = 1$	0.23	63.5	59.5	**15.2**	2.51
$Max = 2$	0.29	65.6	**58.3**	15.4	**2.56**
$Max = 3$	0.33	65.6	60.1	16.2	2.43
$Max = \infty$	0.55	**66.7**	65.7	18.1	2.21

Ablation on Maximum Intervention Limits. To systematically evaluate operational efficiency, we analyzed performance across 96 evaluation episodes on the WidowX platform under varying maximum intervention limits ($Max \in \{1, 2, 3, \infty\}$), as detailed in Table 4. Surprisingly, our evaluation reveals a counterintuitive latency reduction: enabling the World Model for a single retry ($Max = 1$) actually ***decreases the average episode latency*** (15.2s) compared to the baseline (15.7s), despite the approximately 4-second generative overhead. This occurs because the baseline policy often fails to detect errors (e.g., a missed grasp) and ***continues executing*** meaningless actions until its 120-step horizon timeout. In contrast, our critic model acts as an ***early-warning system*** that intervenes to truncate these doomed trajectories. The time saved by reducing the average execution steps (from 67.8 to 59.5) effectively offsets the generative computational cost. To measure real-world viability, we further evaluated the success-per-minute metric, where the $Max = 2$ setting achieves the highest efficiency at 2.56 episodes/min compared to the baseline's 1.92. However, increasing the intervention limit yields diminishing returns. Performance plateaus after increasing to $Max = 3$ yields no additional improvement in success rate (65.6%) but increases latency (16.2s). While an unbounded intervention setting ($Max = \infty$) provides a marginal success gain (66.7%) by resolving a few residual hard cases, it causes a spike in trigger frequency (0.55) and increases average latency to 18.1s. This significant efficiency penalty renders the unbounded configuration less practical for deployment compared to bounded settings.

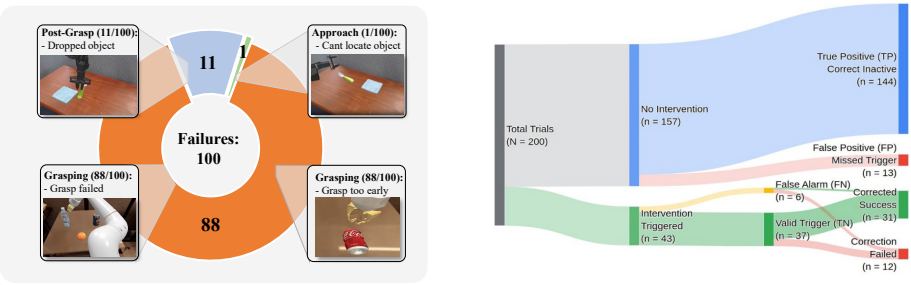
Model Size and Inference Latency. To identify the optimal operating point for real-world deployment, we performed a comprehensive ablation study evaluating the trade-off between model size and inference latency across different configurations (Table 5). In our first configuration (Setting-1), we utilize a 3B Discriminator (0.3s) and a 5B Generator (4s), achieves a +13.5% success rate improvement on the WidowX platform (64.6% compared to the 51.8% baseline) by keeping the intervention cost at approximately 4 seconds. In contrast, Setting-2 configuration scales the models to 7B and 14B, which boosts the success rate improvement to +21.1% by significantly increases the latency to approximately 21 seconds. Although Setting-2 clearly demonstrates the strong scaling potential of our World-in-Loop, the high latency makes it less suitable for real-time execution. Consequently, while larger models highlight the framework's upper

Table 5: Metrics under different model size of discriminator and generator.

	Discriminator (QwenVL-2.5)		Generator (WAN)	
Params.	3B	7B	5B(V2.2)	14B(V2.1)
Latency	0.3s	0.9s	4s	21s
Positive Rate	90.1%	93.1%	\	\
	WidowX	**Google-VM**	**Google-VA**	**Real robot**
Base policy (CogACT)	51.1%	74.8%	61.3%	63.6%
Setting-1	3B		5B	
	64.6%(+**13.5**)	78.5%(+**3.7**)	67.0%(+**5.7**)	76.1%(+**12.5**)
Setting-2 (Large Model size)	7B		14B	
	72.9%(+**21.1**)	78.9%(+**4.1**)	68.6%(+**7.3**)	81.3%(+**17.7**)

performance limits, we consider Setting-1 the optimal balance for real-time deployment, which justifies its selection for the main experiments.

Failure Mode Analysis and Trigger Mechanism Validation. To further understand the failure patterns of the base policy, we conducted a statistical analysis of 100 failed episodes across three platforms (WidowX, Google Robot, and Real Robot). The distribution of failure stages is summarized in Fig. 4.a. Notably, failures during the approach phase are remarkably rare, accounting for only 1% of the total. This indicates that VLA policies are generally robust at identifying the target and reaching its vicinity, and triggering interventions during approach would yield diminishing returns. In contrast, the vast majority of failures (88%) occur precisely at the moment of grasping (e.g., gripper misalignment or timing errors), which strongly validates our framework's design: triggering the world model intervention specifically at the keyframe. The remaining 11% of failures happen during the transport phase after grasping. We honestly acknowledge that our current vision-based framework cannot resolve these irreversible physical failures (e.g., object drop) because pure vision-based system sometimes blocks the object view and lacks the tactile or force feedback necessary to perceive contact loss. Addressing these dynamic failures during transport is the necessary next step for robust manipulation, which is presented in Sec. 5.



a. Distribution of Failure Stage **b. Analysis of Trigger Mechanism**

Fig. 4: The analysis of failure mode and trigger effectiveness of the World Model.

Analysis of WM Trigger Frequency and Effectiveness. We conduct a comprehensive statistical analysis of the system activation frequency of our correction loop. As detailed in Fig. 4.b, we randomly analyzed 200 trials across multi-task benchmarks using our event-triggered loop on the CogACT base policy. The results demonstrate that our World Model trigger mechanism is highly selective and efficient. In our evaluation, the correction loop was activated in only 21.5% of total episodes (FN+TN=43/200), meaning that in the remaining 78.5% of cases, the base policy executed successfully without any extra latency or interruption. Furthermore, the valid trigger rate, representing the necessity of interventions, was notably high, achieving 86% ($TN/(TN + FN) = 37/43$), where the True Negative presents the discriminator correctly predicted an inconsistency with base policy and triggered a correction.

Table 6: Performance comparison with lightweight corrective adapter.

Settings		Google Robot		Real-world	
		Grasp.avg	Succ.avg	Grasp.avg	Succ.avg
Baseline (CogACT)		61.5	51.1	70.9	63.6
#1	RollBack+CogACT Decoder	63.5	52.1	74.0	65.6
	RollBack+Diffusion-Policy [12]	67.7	56.2	79.2	70.8
Ours [RollBack + WM + IDM]		**74.0**	**64.6**	**85.4**	**76.1**

Value of Imagination (vs. Lightweight Adapter). To isolate the contribution of generated videos, we compare a baseline trajectory decoder and a simple diffusion-based [12] replanning under the same rollback setting. #1 in the table shows that WM-based visual futures leverage pretrained physical priors and provide stronger dense task guidance than action-only replanning.

Table 7: Video generation performance evaluation.

Method	**SSIM↑**	**PSNR↑**	**LPIPS↓**	**FVD↓**
Cosmos-predict [32] (action-cond)	24.95	0.85	0.030	146
Wan2.2 [35] (5B)	21.56	0.81	0.041	163
Wan2.1 [35] (14B)	26.27	0.90	0.026	131

Video Generator Evaluation. To ensure that the video generator produces reliable and semantically coherent generations, we first evaluate its generative quality. Specifically, given a test dataset of image-instruction pairs, the video generator takes an initial image and a high-level task instruction as input, and predicts the video sequence corresponding to the execution of that instruction. The generated frames serve as the primary visual basis for subsequent action

decoding, ensuring high visual fidelity is critical to the overall reliability of the pipeline. As shown in Tab. 7, the fine-tuned world model achieves high-quality generation in both simulation and real-world settings, producing temporally coherent and photorealistic outputs with minimal structural or perceptual distortion. The strong alignment between evaluation metrics of different world models further demonstrates the robust generalization, thereby providing a reliable visual foundation for action decoding.

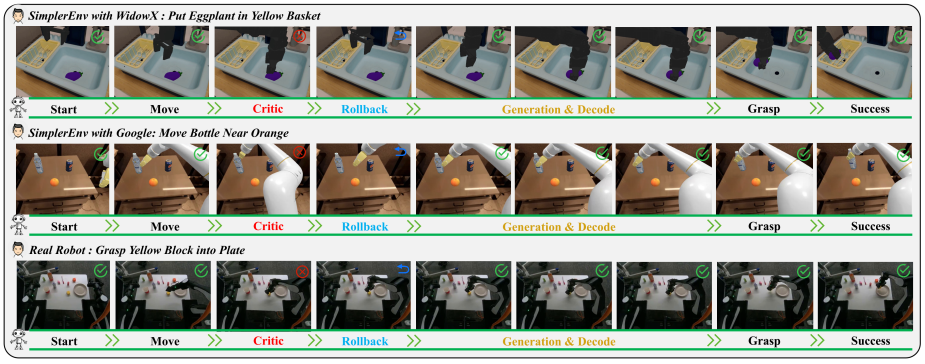


Fig. 5: Qualitative examples of our failure correction process. Each row shows a sequence: (a) the initial failed grasp proposed by the base VLA, (b) the rolled-back state, (c) a keyframe from the imagined successful video generated, and (d) the final successful grasp executed by our framework. Examples are shown for (i) WidowX and (ii) Google Robot in simulation, and the (iii) Xiaomi Robot in the real world.

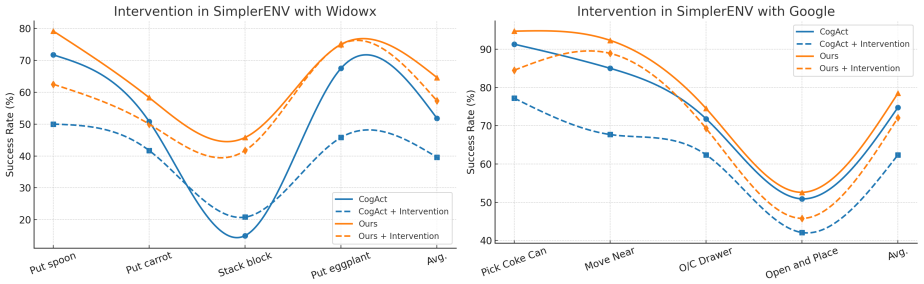


Fig. 6: Robustness to online perturbations in simulation environments.

4.4 Qualitative of Failure Prediction and Correction

We demonstrate our method's efficacy through qualitative failure correction examples and quantitative robustness tests. Fig. 5 illustrates our end-to-end correction pipeline: Where a Critic identifies failed VLA base-grasps to trigger a state rollback, enabling a video generator to imagine successful demonstrations for decoding and executing corrected actions. To evaluate robustness, we introduced controlled perturbations during tasks (Fig. 6). Unlike baselines that either fail or simply halt, our framework maintains a significantly higher success rate under increasing interference. This highlights our framework's superior ability to handle unexpected online disturbances through proactive, generative correction rather than just passive failure detection.

5 Conclusion and Limitation

We introduced World-in-Loop, a novel framework that enhances Vision-Language-Action (VLA) models by integrating a lightweight, event-driven composite World Model (WM) for online policy correction. By repurposing the WM as an active "corrector" rather than a passive predictor, our system triggers discriminative evaluation and generative "imagination" only at critical grasp keyframes, significantly reducing the computational burden of continuous rollouts. The future research will aim to expand this self-correction paradigm into a general-purpose policy-agnostic framework capable of stabilizing a broader range of complex, long-horizon manipulation tasks beyond VLA-based policy.

Clarify on Generalization Beyond Grasping. In this manuscript, we intentionally focused on grasping tasks to validate the core Propose-Evaluate-Imagine-Correct loop, as grasping is a fundamental precursor to the majority of embodied manipulation tasks (e.g., folding clothes requires an initial grasp). However, it is crucial to clarify that our framework is not inherently limited to this domain. Our experimental evaluations already encompass a diverse range of manipulation skills, including ***moving, pushing, stacking, picking, and opening/closing*** tasks, the framework is agnostic to the specific action type and can extend to any task involving distinct state transitions. Generalizing this framework to other manipulations like folding clothes is a very exciting and logical next step, as noted in our future work.

Acknowledgements

This work was funded in part by the Science and Technology Development Fund, Macau SAR (Grant no. 0092/2024/AFJ, 0003/2023/RIB1, 001/2024/SKL), in part by the National Natural Science Foundation of China under Grant 62461160 260, in part by the University of Macau (Grant No.: MYRG-GRG2024-00299-FST-UMDF, MYRG-GRG2025-00312-FST), in part by the Hainan Department of Science and Technology (Grant no. GHYF2026042), and in part by the Guangdong Department of Science and Technology (Grant no. 2023A0505030003).

References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)
2. Agarwal, N., Ali, A., Bala, M., Balaji, Y., Barker, E., Cai, T., Chattopadhyay, P., Chen, Y., Cui, Y., Ding, Y., et al.: Cosmos world foundation model platform for physical ai. arXiv preprint arXiv:2501.03575 (2025)
3. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., et al.: Qwen2. 5-vl technical report. arXiv preprint arXiv:2502.13923 (2025)
4. Brohan, A., Brown, N., Carbajal, J., Chebotar, Y., Dabis, J., Finn, C., Gopalakrishnan, K., Hausman, K., Herzog, A., Hsu, J., et al.: Rt-1: Robotics transformer for real-world control at scale. arXiv preprint arXiv:2212.06817 (2022)
5. Cen, J., Yu, C., Yuan, H., Jiang, Y., Huang, S., Guo, J., Li, X., Song, Y., Luo, H., Wang, F., et al.: Worldvla: Towards autoregressive action world model. arXiv preprint arXiv:2506.21539 (2025)
6. Chen, Q., Li, F., Xu, S., Lai, Z., Xie, Z., Luo, Y., Jiang, S., Li, H., Chen, L., Wang, B., et al.: Vilta: A vlm-in-the-loop adversary for enhancing driving policy robustness. arXiv preprint arXiv:2601.12672 (2026)
7. Ebert, F., Finn, C., Dasari, S., Xie, A., Lee, A., Levine, S.: Visual foresight: Model-based deep reinforcement learning for vision-based robotic control. arXiv preprint arXiv:1812.00568 (2018)
8. Ha, D., Schmidhuber, J.: World models. arXiv preprint arXiv:1803.10122 (2018)
9. Hafner, D., Lillicrap, T., Ba, J., Norouzi, M.: Dream to control: Learning behaviors by latent imagination. In: International Conference on Learning Representations (ICLR) (2020)
10. Hafner, D., Lillicrap, T., Fischer, I., Villegas, R., Ha, D., Lee, H., Davidson, J.: Learning latent dynamics for planning from pixels. In: International conference on machine learning. pp. 2555–2565. PMLR (2019)
11. Hao, X., Zhou, L., Huang, Z., Hou, Z., Tang, Y., Zhang, L., Li, G., Lu, Z., Ren, S., Meng, X., et al.: Mimo-embodied: X-embodied foundation model technical report. arXiv preprint arXiv:2511.16518 (2025)
12. Intelligence, P., Black, K., Brown, N., Darpinian, J., Dhabalia, K., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., et al.: $\pi_{0.5}$: a vision-language-action model with open-world generalization. arXiv preprint arXiv:2504.16054 (2025)
13. Jiang, Y., Gupta, A., Zhang, Z., Wang, G., Dou, Y., Chen, Y., Fei-Fei, L., Anandkumar, A., Zhu, Y., Fan, L.: Vima: General robot manipulation with multimodal prompts. arXiv preprint arXiv:2210.03094 **2**(3), 6 (2022)
14. Kahn, G., Villaflor, A., Pong, V., Abbeel, P., Levine, S.: Uncertainty-aware reinforcement learning for collision avoidance. arXiv preprint arXiv:1702.01182 (2017)
15. Kim, M.J., Gao, Y., Lin, T.Y., Lin, Y.C., Ge, Y., Lam, G., Liang, P., Song, S., Liu, M.Y., Finn, C., et al.: Cosmos policy: Fine-tuning video models for visuomotor control and planning. arXiv preprint arXiv:2601.16163 (2026)
16. Kim, M.J., Pertsch, K., Karamcheti, S., Xiao, T., Balakrishna, A., Nair, S., Rafailov, R., Foster, E., Lam, G., Sanketi, P., et al.: Openvla: An open-source vision-language-action model. arXiv preprint arXiv:2406.09246 (2024)
17. Li, B., Zhang, Y., Guo, D., Zhang, R., Li, F., Zhang, H., Zhang, K., Zhang, P., Li, Y., Liu, Z., et al.: Llava-onevision: Easy visual task transfer. arXiv preprint arXiv:2408.03326 (2024)

18. Li, P., Wu, K., Xu, S., Li, F., Li, H., Zhao, L., Lyu, K., Chen, L., Yang, Z.X., Zheng, N.: Spaact: Spatially-activated transition learning with curriculum adaptation for vision-language navigation. arXiv preprint arXiv:2604.27620 (2026)
19. Li, Q., Liang, Y., Wang, Z., Luo, L., Chen, X., Liao, M., Wei, F., Deng, Y., Xu, S., Zhang, Y., et al.: Cogact: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation. arXiv preprint arXiv:2411.19650 (2024)
20. Li, X., Li, P., Liu, M., Wang, D., Liu, J., Kang, B., Ma, X., Kong, T., Zhang, H., Liu, H.: Towards generalist robot policies: What matters in building vision-language-action models. arXiv preprint arXiv:2412.14058 (2024)
21. Li, X., Hsu, K., Gu, J., Pertsch, K., Mees, O., Walke, H.R., Fu, C., Lunawat, I., Sieh, I., Kirmani, S., Levine, S., Wu, J., Finn, C., Su, H., Vuong, Q., Xiao, T.: Evaluating real-world robot manipulation policies in simulation. In: Proceedings of the Conference on Robot Learning (CoRL) (2024)
22. Liu, B., Zhu, Y., Gao, C., Feng, Y., Liu, Q., Zhu, Y., Stone, P.: Libero: Benchmarking knowledge transfer for lifelong robot learning. Advances in Neural Information Processing Systems **36**, 44776–44791 (2023)
23. Luo, Y., Li, F., Xu, S., Ji, Y., Zhang, Z., Wang, B., Shen, Y., Cui, J., Chen, L., Chen, G., et al.: Last-vla: Thinking in latent spatio-temporal space for vision-language-action in autonomous driving. arXiv preprint arXiv:2603.01928 (2026)
24. Luo, Y., Li, F., Xu, S., Lai, Z., Yang, L., Chen, Q., Luo, Z., Xie, Z., Jiang, S., Liu, J., et al.: Adathinkdrive: Adaptive thinking via reinforcement learning for autonomous driving. arXiv preprint arXiv:2509.13769 (2025)
25. Luo, Z., Yang, Y., Cai, C., Zhang, Y., Zheng, F.: Roboreflect: Robotic reflective reasoning for grasping ambiguous-condition objects. arXiv preprint arXiv:2501.09307 (2025)
26. O'Neill, A., Rehman, A., Maddukuri, A., Gupta, A., Padalkar, A., Lee, A., Pooley, A., Gupta, A., Mandlekar, A., Jain, A., et al.: Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In: 2024 IEEE International Conference on Robotics and Automation (ICRA). pp. 6892–6903. IEEE (2024)
27. Pai, J., Achenbach, L., Montesinos, V., Forrai, B., Mees, O., Nava, E.: mimic-video: Video-action models for generalizable robot control beyond vlas. arXiv preprint arXiv:2512.15692 (2025)
28. Qu, D., Song, H., Chen, Q., Yao, Y., Ye, X., Ding, Y., Wang, Z., Gu, J., Zhao, B., Wang, D., et al.: Spatialvla: Exploring spatial representations for visual-language-action model. arXiv preprint arXiv:2501.15830 (2025)
29. Shi, H., Xie, B., Liu, Y., Sun, L., Liu, F., Wang, T., Zhou, E., Fan, H., Zhang, X., Huang, G.: Memoryvla: Perceptual-cognitive memory in vision-language-action models for robotic manipulation. arXiv preprint arXiv:2508.19236 (2025)
30. Shi, L.X., Hu, Z., Zhao, T.Z., Sharma, A., Pertsch, K., Luo, J., Levine, S., Finn, C.: Yell at your robot: Improving on-the-fly from language corrections. arXiv preprint arXiv:2403.12910 (2024)
31. Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., Yao, S.: Reflexion: Language agents with verbal reinforcement learning. Advances in Neural Information Processing Systems **36**, 8634–8652 (2023)
32. Team, N.: World simulation with video foundation models for physical ai (2025), `https://arxiv.org/abs/2511.00062`
33. Team, O.M., Ghosh, D., Walke, H., Pertsch, K., Black, K., Mees, O., Dasari, S., Hejna, J., Kreiman, T., Xu, C., et al.: Octo: An open-source generalist robot policy. arXiv preprint arXiv:2405.12213 (2024)

34. Walke, H.R., Black, K., Zhao, T.Z., Vuong, Q., Zheng, C., Hansen-Estruch, P., He, A.W., Myers, V., Kim, M.J., Du, M., et al.: Bridgedata v2: A dataset for robot learning at scale. In: Conference on Robot Learning. pp. 1723–1736. PMLR (2023)
35. Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.W., Chen, D., Yu, F., Zhao, H., Yang, J., et al.: Wan: Open and advanced large-scale video generative models. arXiv preprint arXiv:2503.20314 (2025)
36. Wu, J., Yin, S., Feng, N., He, X., Li, D., Hao, J., Long, M.: ivideogpt: Interactive videogpts are scalable world models. Advances in Neural Information Processing Systems **37**, 68082–68119 (2025)
37. Xia, W., Feng, R., Wang, D., Hu, D.: Phoenix: A motion-based self-reflection framework for fine-grained robotic action correction. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 6981–6990 (2025)
38. Xiao, J., Yang, Y., Chang, X., Chen, R., Xiong, F., Xu, M., Zheng, W.S., Zhang, Q.: World-env: Leveraging world model as a virtual environment for vla post-training. arXiv preprint arXiv:2509.24948 (2025)
39. Yang, Y., Duan, Z., Xie, T., Cao, F., Shen, P., Song, P., Zhao, C., Jin, P., Sun, G., Xu, S., et al.: Fpc-vla: A vision-language-action framework with a supervisor for failure prediction and correction. Expert Systems with Applications p. 131742 (2026)
40. Zheng, J., Li, J., Wang, Z., Liu, D., Kang, X., Feng, Y., Zheng, Y., Zou, J., Chen, Y., Zeng, J., et al.: X-vla: Soft-prompted transformer as scalable cross-embodiment vision-language-action model. arXiv preprint arXiv:2510.10274 (2025)
41. Zheng, R., Liang, Y., Huang, S., Gao, J., Daumé III, H., Kolobov, A., Huang, F., Yang, J.: Tracevla: Visual trace prompting enhances spatial-temporal awareness for generalist robotic policies. arXiv preprint arXiv:2412.10345 (2024)
42. Zhu, F., Yan, Z., Hong, Z., Shou, Q., Ma, X., Guo, S.: Wmpo: World model-based policy optimization for vision-language-action models. arXiv preprint arXiv:2511.09515 (2025)
43. Zhu, J., Wang, W., Chen, Z., Liu, Z., Ye, S., Gu, L., Tian, H., Duan, Y., Su, W., Shao, J., et al.: Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. arXiv preprint arXiv:2504.10479 (2025)
44. Zitkovich, B., Yu, T., Xu, S., Xu, P., Xiao, T., Xia, F., Wu, J., Wohlhart, P., Welker, S., Wahid, A., et al.: Rt-2: Vision-language-action models transfer web knowledge to robotic control. In: Conference on Robot Learning. pp. 2165–2183. PMLR (2023)