CAG-WAM: CAUSAL ACTION-GAIN WORLD ACTION MODELS FOR ROBOT MANIPULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

World Action Models (WAMs) combine action generation with future prediction, allowing predicted futures to support action evaluation and robot decision making. However, the future associated with a candidate action may contain action-induced and environment-driven changes. Directly evaluating the predicted future may not fully characterize the future change associated with the candidate action. For example, a candidate action that worsens a favorable environmental future may still yield a favorable outcome, while one that improves an unfavorable environmental future may still yield an unfavorable outcome. To analyze this issue, we formulate a Structural Causal Model (SCM) for action evaluation in WAMs. The SCM formalizes how action-induced and environment-driven changes jointly shape future evaluation, and reveals a backdoor path through which environmental evolution can also affect the future evaluation associated with a candidate action. Intervention and backdoor adjustment account for this path, providing a causal basis for characterizing the future change associated with each candidate action. Based on this analysis, we propose Causal Action-Gain World Action Model (CAG-WAM). For each candidate action, CAG-WAM predicts an action-conditioned future and an environmental future under the same current observation, and defines their evaluation difference as causal action gain (CAG) for characterizing the future change associated with the candidate action. During training, we incorporate CAG into Group Relative Policy Optimization (GRPO) to guide robot policy learning. During inference, we evaluate the initial action with CAG and, when needed, revise it using predicted future information. Experiments on multiple benchmarks support the effectiveness of CAG-WAM.

1 INTRODUCTION

World Action Models (WAMs) have recently been explored for robotic manipulation by combining action generation with future prediction (Ye et al., 2026; Team et al., 2026; Chen et al., 2026). Given current visual and proprioceptive observations together with a task instruction, these models are trained to generate robot actions and predict future observations or scene evolution (Liu et al., 2026b; Pan et al., 2026; Zhu et al., 2026). Such future predictions provide information about the possible outcomes of generated actions and can inform subsequent action generation and decision making. Some recent WAMs further generate multiple candidate actions and predict the future associated with each candidate before execution (Zhou et al., 2026; Xue et al., 2026). These predicted futures can then be evaluated using task-related information, such as task progress or value estimates, to compare candidate actions and guide action selection (Wang et al., 2026b; Lin et al., 2025).

However, the predicted future associated with an action may contain changes from different sources. During robot–environment interaction, some future changes are associated with the action, while others may arise from environmental dynamics or external factors, such as passive object motion or external disturbances (Liang et al., 2026; Jiang et al., 2026; Gu et al., 2026; Wang et al., 2026a). Consequently, the overall future evaluation alone may not fully characterize the future change associated with the action. For example, in **Scenario 1** of **Figure 1a**, the environment evolves toward a favorable future in which the robot remains near the object. Candidate actions that move the robot away can still result in futures that appear favorable overall, despite worsening the state relative to the environmental evolution. Conversely, in **Scenario 2**, the environment evolves toward an unfavorable future in which the robot remains far from the sink. Candidate actions that move the robot

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

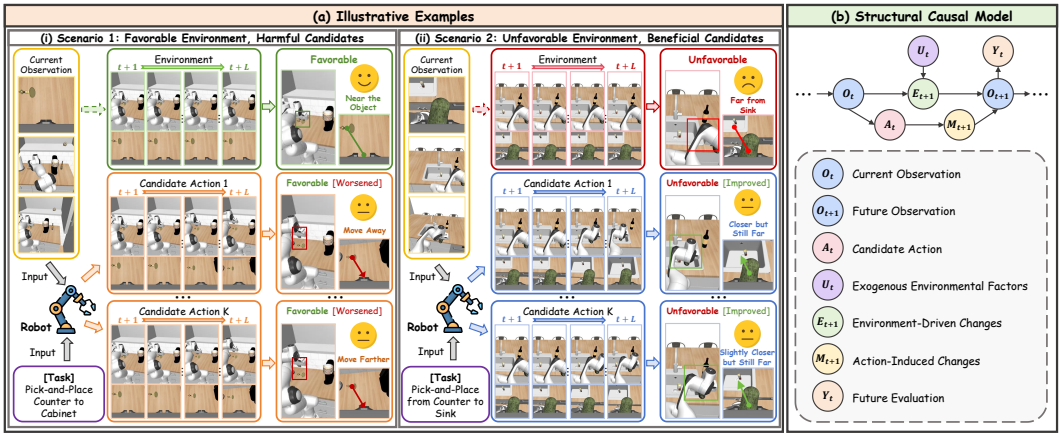


Figure 1: Motivation and causal analysis. (a) Illustrative cases where the overall future evaluation may not fully reflect the future change associated with a candidate action. (b) Structural causal model of action evaluation, including the backdoor path $\mathbf{A}_t \leftarrow \mathbf{O}_t \rightarrow \mathbf{E}_{t+1} \rightarrow \mathbf{O}_{t+1} \rightarrow Y_t$.

closer can still result in futures that appear unfavorable overall, despite improving the state relative to the environmental evolution. A natural approach is to compare multiple candidates generated under the same observation (Zhou et al., 2026; Shao et al., 2024). Such comparison can distinguish relatively better and worse candidates within the current set, but it still evaluates each candidate relative to the other sampled actions. For example, in **Scenario 1**, a relative comparison may prefer the candidate that moves the robot away over the one that moves it farther away, although both candidates worsen the favorable environmental future. In **Scenario 2**, the candidate that moves the robot only slightly closer may be less preferred than the one that moves it farther toward the sink, although both candidates improve the unfavorable environmental future. Thus, we focus on a central question: **how can a WAM characterize the future change associated with each candidate action when the predicted future contains changes from different sources, and use this information to guide policy learning and action generation?**

To address this problem, we analyze action evaluation from a causal perspective. We construct the Structural Causal Model (SCM) shown in **Figure 1b** to describe the relationships among the current observation, candidate action, different sources of future change, and future evaluation. Specifically, the current observation $\mathbf{O}_t$ influences both the generation of a candidate action $\mathbf{A}_t$ and the subsequent environment-driven changes $\mathbf{E}_{t+1}$. Meanwhile, the candidate action gives rise to action-induced changes $\mathbf{M}_{t+1}$, and these two sources of change jointly shape the future observation $\mathbf{O}_{t+1}$, which is then evaluated as Y_t. In addition to its effect through the action-induced changes, $\mathbf{A}_t$ and Y_t are also connected through the backdoor path $\mathbf{A}_t \leftarrow \mathbf{O}_t \rightarrow \mathbf{E}_{t+1} \rightarrow \mathbf{O}_{t+1} \rightarrow Y_t$ (Pearl, 2009; Peters et al., 2017). This path reflects that the current observation affects both which action is generated and how the environment may evolve. Therefore, the future evaluation associated with an action may reflect both action-induced and environment-driven changes, making it difficult to characterize the future change associated with the action from the overall future evaluation alone. Backdoor adjustment provides a principled way to account for this backdoor path. Specifically, the intervention $do(\mathbf{A}_t = \mathbf{a})$ fixes the action to $\mathbf{a}$ and cuts the incoming edge $\mathbf{O}_t \rightarrow \mathbf{A}_t$ in the causal graph, thereby blocking this path (Pearl, 2009; Hernán & Robins, 2010). This analysis therefore provides a causal basis for characterizing the future change associated with each candidate action.

Motivated by this analysis, we propose Causal Action-Gain World Action Model (CAG-WAM), which characterizes the future change associated with each candidate action and uses this information to guide policy learning and action generation. CAG-WAM consists of a Future-Conditioned Action Model (FC-AM) and a Paired-Future World Model (PF-WM), with a shared future evaluator used to evaluate the predicted futures. Given the current observation and task instruction, the FC-AM generates candidate action chunks, while the PF-WM predicts an action-conditioned future for each candidate and an environmental future from the same current observation. The action-conditioned future describes the predicted outcome under the candidate action, while the environmental future describes how the scene may evolve without applying that action. The two futures are evaluated

by the same future evaluator, and their evaluation difference defines the causal action gain (CAG), which characterizes the future change associated with the candidate action relative to environmental evolution. During training, we use CAG as the advantage in group relative policy optimization (GRPO) (Shao et al., 2024), so that policy updates can reflect the future change associated with each candidate action. During inference, the initial action is evaluated using the same paired-future comparison. When its CAG falls below the revision threshold, future information from the action-conditioned future and its difference from the environmental future is provided to the same action model for a second generation. The revised action is then combined with the initial action before execution. In this way, CAG-WAM can emphasize the future change associated with each candidate action in both policy optimization during training and action revision during inference. Extensive experiments support the effectiveness of CAG-WAM.

The main contributions of this paper are summarized as follows: (**i**) We formulate action evaluation in WAMs from a causal perspective and construct a structural causal model that describes how action-induced and environment-driven changes jointly affect future evaluation and identifies the associated backdoor path. (**ii**) We propose CAG-WAM, which introduces causal action gain to characterize the future change associated with each candidate action relative to environmental evolution, and uses this information to guide policy learning and action revision. (**iii**) Experiments conducted on multiple benchmarks support the effectiveness of CAG-WAM.

2 Related Work

World Action Models (WAMs) integrate future prediction with robot action generation, allowing predicted outcomes to support manipulation decisions. Motubrain (Team et al., 2026) jointly models video dynamics and robot actions within a unified generative framework. τ_0-WM (Zhou et al., 2026) evaluates candidate actions through action-conditioned future prediction and task-progress estimation, and uses predicted futures for action refinement. SelfWAM (Pan et al., 2026) strengthens the dependence of predicted futures on executed actions through explicit action conditioning and self-mask supervision. HiTac-WAM (Xue et al., 2026) incorporates predicted tactile futures for candidate-action ranking in contact-rich manipulation. These studies model action-conditioned futures and use predicted outcomes for candidate evaluation, action selection, or refinement. Our work instead focuses on action-conditioned future evaluation from a causal perspective, considering that the predicted future may contain changes associated with both the candidate action and environmental evolution. We therefore characterize the future change associated with each candidate action to guide policy learning and action generation.

Causal reasoning has been applied to robot learning through interventions, counterfactual reasoning, and deconfounding to study the relationships among observations, actions, and task outcomes. CREST (Lee et al., 2021) uses interventions in internal models to identify task-relevant state variables for robot manipulation policies. RoCoDA (Ameperosa et al., 2025) exploits causal invariance to construct counterfactual demonstrations by modifying task-irrelevant environmental factors. CofactVLA (Zhang et al., 2026) models visual confounding in action generation by vision-language-action (VLA) models and introduces counterfactual intervention to reduce spurious visual dependencies. FACT (Peng et al., 2026) introduces causal training into WAMs by conditioning future video and task-progress prediction on executed actions and incorporating failure rollouts to model action-dependent consequences. These studies apply causal reasoning to different aspects of robot learning, including state-variable identification, counterfactual data construction, action generation, and future prediction. Our work focuses on future evaluation in WAMs and constructs a structural causal model to analyze how candidate actions and environmental evolution jointly influence future evaluation. Based on this analysis, we characterize the future change associated with each candidate action, rather than relying on the overall future evaluation alone, and use this information to guide policy learning and action generation.

3 Problem Settings and Analysis

Problem Setting We consider a World Action Model (WAM) for robotic manipulation that supports both action generation and future prediction (Ye et al., 2026; Team et al., 2026; Zhou et al., 2026).

At time step t, let $\mathbf{O}_t$ denote the observations available to the robot, including visual and proprioceptive observations. Given the current observation $\mathbf{O}_t$ and task instruction $\mathbf{c}$, the action policy samples K candidate actions $\mathbf{A}_t^i \sim \pi_\theta(\cdot|\mathbf{O}_t, \mathbf{c})$, where $i = 1, \ldots, K$ and θ denotes the policy parameters. For each candidate action, the world model predicts its corresponding future as $\hat{\mathbf{O}}_{t+1:t+H}^i \sim p_\phi(\cdot|\mathbf{O}_t, \mathbf{A}_t^i, \mathbf{c})$, where $\hat{\mathbf{O}}_{t+1:t+H}^i$ denotes the predicted future observation sequence over a prediction horizon H, and ϕ denotes the world-model parameters.

We evaluate each candidate's predicted future as $Y_t^i = \mathcal{V}(\hat{\mathbf{O}}_{t+1:t+H}^i, \mathbf{c})$, where $\mathcal{V}(\cdot)$ denotes a generic future evaluator that returns a scalar task-related score, such as a task reward, value estimate, task-progress score, or a combination thereof. The resulting future evaluations can then be used for candidate comparison, action selection, or subsequent policy optimization.

Motivation Analysis WAMs can use the future evaluations Y_t^i defined above for candidate comparison and action selection (Zhou et al., 2026; Xue et al., 2026; Wang et al., 2026b; Lin et al., 2025). However, the future predicted for a candidate action may reflect changes from multiple sources. During robot–environment interaction, some changes may be associated with the robot action, while others may arise from environmental dynamics or external factors, such as passive object motion or external disturbances (Liang et al., 2026; Jiang et al., 2026; Gu et al., 2026; Wang et al., 2026a). Consequently, the evaluation of the resulting future may not by itself characterize the change associated with the candidate action. As illustrated in **Figure 1a**, **Scenario 1** considers a favorable environment in which the robot would remain near the target object without applying the candidate action. A candidate action that moves the robot away may still lead to a future that appears favorable overall, although it worsens the state relative to the environmental future. In contrast, **Scenario 2** considers an unfavorable environment in which the robot would remain far from the sink without applying the candidate action. A candidate action that moves the robot closer may still lead to a future that appears unfavorable overall, although it improves the state relative to the environmental future. These cases show that the overall evaluation of the predicted future and the change associated with the candidate action may not be equivalent.

A natural approach is to compare multiple candidates generated from the same current observation (Zhou et al., 2026; Shao et al., 2024). Such comparisons rank the sampled candidates according to their future evaluations: $Y_t^i > Y_t^j$ favors candidate i over candidate j. **Figure 1a** illustrates the limitation of relying on this relative ranking alone. In **Scenario 1**, the candidate that moves the robot away can be preferred over the candidate that moves it farther away, although both actions worsen the favorable environmental future. Conversely, in **Scenario 2**, the candidate that moves the robot only slightly closer can be less preferred than the candidate that moves it farther toward the sink, although both actions improve the unfavorable environmental future. Thus, candidate-relative comparison indicates which sampled actions are relatively preferred, but does not by itself characterize whether each individual action improves or worsens the future relative to environmental evolution. Therefore, we focus on a central question: **how can a WAM characterize the future change associated with each candidate action when the predicted future contains changes from multiple sources, and use this information to guide policy learning and action generation?**

Causal Analysis To address this issue, we analyze action evaluation in WAMs from a causal perspective. We construct a Structural Causal Model (SCM) to describe the relationships among the current observation, candidate action, different sources of future change, and future evaluation.

As shown in **Figure 1b**, let $\mathbf{O}_t$ denote the current observation, $\mathbf{A}_t$ a candidate action, and $\mathbf{U}_t$ exogenous environmental factors that are not directly controlled by the current action. We use $\mathbf{E}_{t+1}$ to denote environment-driven changes and $\mathbf{M}_{t+1}$ to denote action-induced changes. The environment-driven changes $\mathbf{E}_{t+1}$ depend on the current observation $\mathbf{O}_t$ and exogenous environmental factors $\mathbf{U}_t$, while the candidate action $\mathbf{A}_t$ gives rise to the action-induced changes $\mathbf{M}_{t+1}$. These two sources of change jointly shape the future observation $\mathbf{O}_{t+1}$, which is subsequently evaluated as Y_t under the task instruction $\mathbf{c}$. Thus, the predicted future may contain changes arising from both the candidate action and environmental evolution.

Because the candidate action $\mathbf{A}_t$ is generated conditioned on the current observation $\mathbf{O}_t$, $\mathbf{O}_t$ influences which action is generated and also affects the future through the environment-driven changes $\mathbf{E}_{t+1}$. The candidate action $\mathbf{A}_t$ affects Y_t through the action-induced changes $\mathbf{M}_{t+1}$, while $\mathbf{A}_t$ and Y_t are also connected through the backdoor path $\mathbf{A}_t \leftarrow \mathbf{O}_t \rightarrow \mathbf{E}_{t+1} \rightarrow \mathbf{O}_{t+1} \rightarrow Y_t$. This path

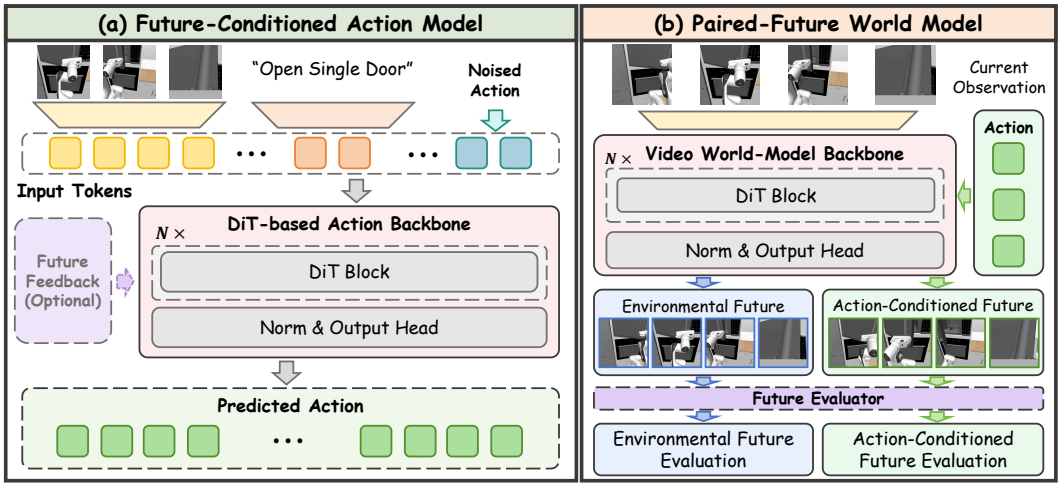


Figure 2: CAG-WAM Architecture. (a) Future-Conditioned Action Model generates action chunks from the current observation and task instruction, and can additionally incorporate future feedback for action revision. (b) Paired-Future World Model predicts action-conditioned and environmental futures under different action conditions and evaluates them using the same future evaluator.

reflects that the same current observation influences both which action is generated and how the environment may evolve. Consequently, the future evaluation associated with a candidate action may reflect both action-induced and environment-driven changes, making it difficult to characterize the future change associated with the action from the overall future evaluation alone. For clarity, we condition on the task instruction $\mathbf{c}$ throughout this analysis and omit it from the causal notation. Formally, the expected future evaluation associated with action $\mathbf{a}$ can be written as

$$\mathbb{E}[Y_t \mid \mathbf{A}_t = \mathbf{a}] = \int \mathbb{E}[Y_t \mid \mathbf{A}_t = \mathbf{a}, \mathbf{O}_t = \mathbf{o}] p(\mathbf{o} \mid \mathbf{A}_t = \mathbf{a}) d\mathbf{o}, \tag{1}$$

where $\mathbf{a}$ and $\mathbf{o}$ denote possible realizations of $\mathbf{A}_t$ and $\mathbf{O}_t$, respectively. Because candidate actions are generated from $\mathbf{O}_t$, $p(\mathbf{o}|\mathbf{A}_t = \mathbf{a})$ can vary with the action. Since $\mathbf{O}_t$ also affects Y_t through $\mathbf{E}_{t+1}$, this evaluation may also reflect the influence of the backdoor path.

Backdoor adjustment provides a principled way to account for this path through intervention on the action variable (Pearl, 2009; Hernán & Robins, 2010). Specifically, the intervention $do(\mathbf{A}_t = \mathbf{a})$ fixes the action to $\mathbf{a}$ and cuts the incoming edge $\mathbf{O}_t \to \mathbf{A}_t$ in the causal graph, thereby blocking the backdoor path. Under the SCM, the corresponding interventional evaluation can be expressed as

$$\mathbb{E}[Y_t \mid do(\mathbf{A}_t = \mathbf{a})] = \int \mathbb{E}[Y_t \mid \mathbf{A}_t = \mathbf{a}, \mathbf{O}_t = \mathbf{o}] p(\mathbf{o}) d\mathbf{o}. \tag{2}$$

Compared with **Eq. 1**, the action-dependent observation distribution $p(\mathbf{o}|\mathbf{A}_t = \mathbf{a})$ is replaced by the observation distribution $p(\mathbf{o})$, thereby reducing the influence associated with the backdoor path and allowing the action evaluation to better reflect the future change associated with the action.

Overall, this causal analysis shows that action-induced and environment-driven changes jointly shape future evaluation, while the SCM reveals a backdoor path connecting the candidate action and future evaluation through the current observation and environment-driven changes. By accounting for this path through intervention and backdoor adjustment, this analysis provides a causal basis for characterizing the future change associated with each candidate action.

4 METHODOLOGY

In this section, we propose Causal Action-Gain World Action Model (CAG-WAM) to characterize the future change associated with each candidate action and use this information to guide both policy learning and action generation. **Section 4.1** introduces the overall architecture of CAG-WAM and the interaction among action generation, future prediction, and future evaluation. **Section 4.2**

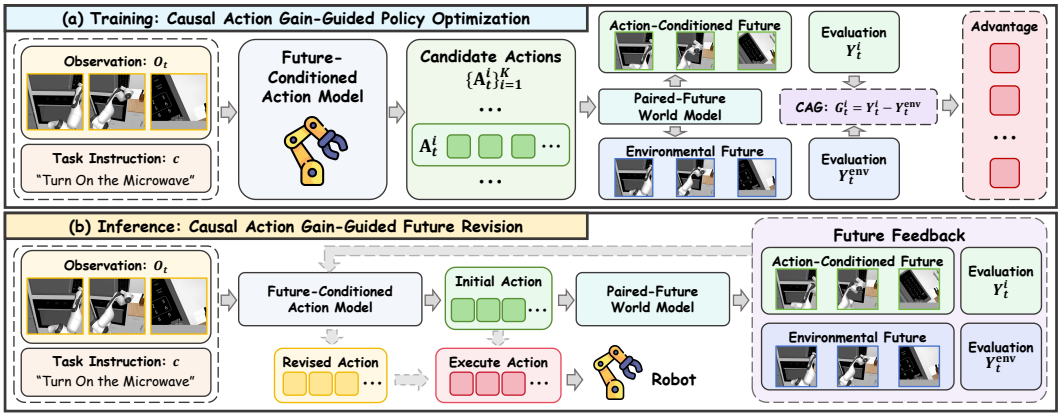


Figure 3: Training and inference framework of CAG-WAM. (a) Training: candidate actions are evaluated by comparing their action-conditioned futures with the environmental future, and the resulting causal action gain is used as the policy-optimization advantage. (b) Inference: the initial action is evaluated using the same paired-future comparison; when its causal action gain falls below the revision threshold, future feedback is used for an additional action generation before execution.

presents how causal action gain is computed and incorporated into policy optimization. **Section 4.3** describes how the predicted future information is further used to revise actions during inference.

4.1 Architecture: Causal Action-Gain World Action Model

As illustrated in **Figure 2**, CAG-WAM consists of a Future-Conditioned Action Model (FC-AM) and a Paired-Future World Model (PF-WM). (i) The FC-AM adopts a flow-matching formulation with a DiT backbone. Given the current observation $\mathbf{O}_t$ and task instruction $\mathbf{c}$, it generates K candidate action chunks $\{\mathbf{A}_t^i\}_{i=1}^K$, where $\mathbf{A}_t^i = [\mathbf{a}_t^i, \dots, \mathbf{a}_{t+L-1}^i]$ and L denotes the action-chunk length. Besides generating actions from the current observation, the same FC-AM can also incorporate predicted future information as an additional condition for subsequent action generation. (ii) The PF-WM is an action-conditioned video world model that predicts future observations under different action conditions. For each candidate action $\mathbf{A}_t^i$, it predicts the corresponding action-conditioned future $\hat{\mathbf{O}}_{t+1:t+H}^i$, where H denotes the prediction horizon. From the same current observation, the PF-WM also predicts an environmental future $\hat{\mathbf{O}}_{t+1:t+H}^{\text{env}}$, describing how the scene may evolve without applying the candidate action. Each action-conditioned future and the environmental future are generated from the same current observation and task instruction, while differing in their action conditions. They are evaluated by the same future evaluator, yielding $Y_t^i = \mathcal{V}(\hat{\mathbf{O}}_{t+1:t+H}^i, \mathbf{c})$ and $Y_t^{\text{env}} = \mathcal{V}(\hat{\mathbf{O}}_{t+1:t+H}^{\text{env}}, \mathbf{c})$. Their evaluation difference is used to guide policy optimization during training, while information from the predicted futures is used to condition the FC-AM for action revision during inference.

4.2 Training: Causal Action Gain-Guided Policy Optimization

In this section, we introduce how the action policy is trained using causal action gain. We first compute the causal action gain of each candidate by comparing its predicted future with the environmental future. We then incorporate the causal action gain into GRPO, allowing the future change associated with each candidate action to guide policy learning.

4.2.1 Causal Action Gain

Given the K candidate action chunks $\{\mathbf{A}_t^i\}_{i=1}^K$ generated from the current observation $\mathbf{O}_t$, we characterize the future change associated with each candidate through paired future prediction. For each candidate action $\mathbf{A}_t^i$, the PF-WM predicts its action-conditioned future $\hat{\mathbf{O}}_{t+1:t+H}^i$, which describes the predicted outcome under the candidate action. From the same current observation, the PF-WM also predicts an environmental future $\hat{\mathbf{O}}_{t+1:t+H}^{\text{env}}$, which describes how the scene may evolve without

applying the candidate action. The action-conditioned future and environmental future are generated from the same current observation and task instruction, providing a matched comparison for evaluating the change associated with the candidate action.

We evaluate both futures using the same future evaluator $\mathcal{V}$: $Y_t^i = \mathcal{V}(\hat{\mathbf{O}}^i_{t+1:t+H}, \mathbf{c})$ and $Y_t^{\text{env}} = \mathcal{V}(\hat{\mathbf{O}}^{\text{env}}_{t+1:t+H}, \mathbf{c})$. Here, Y_t^i and Y_t^{env} denote the evaluations of the action-conditioned future and environmental future, respectively. Following **Section 3**, we define the causal action gain of candidate action $\mathbf{A}_t^i$ as $G_t^i = Y_t^i - Y_t^{\text{env}}$. A positive G_t^i means that the action-conditioned future receives a higher evaluation than the environmental future, while a negative value means that it receives a lower evaluation. The magnitude of G_t^i reflects the evaluation difference between the two futures. Thus, CAG uses the paired future evaluations to characterize the future change associated with the candidate action relative to environmental evolution (see **Appendix C** for further discussion).

4.2.2 Causal Action Gain-Guided Optimization

After obtaining the causal action gain, we use it to optimize the action policy. Group Relative Policy Optimization (GRPO) (Shao et al., 2024) provides an efficient way to optimize policies from grouped samples without requiring an additional critic model, and has recently been extended to embodied reasoning and robot policy learning (Liu et al., 2026a; Zhao et al., 2025; Cao et al., 2026). We incorporate the causal action gain into GRPO so that the future change associated with each candidate action can directly inform policy optimization.

Specifically, given the current observation $\mathbf{O}_t$ and task instruction $\mathbf{c}$, the flow-based action policy samples K candidate action chunks $\{\mathbf{A}_t^i\}_{i=1}^K$. For each candidate action chunk $\mathbf{A}_t^i$, the paired future evaluation provides a causal action gain G_t^i. We use this gain as the advantage of the corresponding candidate action, i.e., $\Delta^i_{t,\text{CAG}} = G_t^i$.

We then optimize the action policy using the clipped GRPO objective:

$$\mathcal{J}_{\text{CAG}}(\theta) = \mathbb{E}\left[\frac{1}{K}\sum_{i=1}^{K}\frac{1}{T}\sum_{\tau=1}^{T}(\min(\rho^i_{t,\tau}(\theta)G_t^i, \text{clip}(\rho^i_{t,\tau}(\theta), 1-\epsilon, 1+\epsilon)G_t^i) - \beta D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}}))\right] \quad (3)$$

where T denotes the number of flow-generation steps, ϵ is the clipping coefficient, β controls KL regularization, and π_{ref} denotes the reference policy. Let $\mathbf{X}^i_{t,\tau}$ denote the intermediate action variable of candidate i at generation step τ. The importance ratio between the current policy and the sampling policy is computed as $\rho^i_{t,\tau}(\theta) = \frac{p_\theta(\mathbf{X}^i_{t,\tau-1}|\mathbf{X}^i_{t,\tau},\mathbf{O}_t,\mathbf{c})}{p_{\theta_{\text{old}}}(\mathbf{X}^i_{t,\tau-1}|\mathbf{X}^i_{t,\tau},\mathbf{O}_t,\mathbf{c})}$ following the probabilistic formulation for flow-based policy optimization (Liu et al., 2026a) (see **Appendix B.3** for more details).

In this way, candidate action chunks with positive causal action gains tend to be encouraged, whereas those with negative gains tend to be suppressed. The future change associated with each candidate action, as characterized by the causal action gain, can guide the optimization of the action policy.

4.3 Inference: Causal Action Gain-Guided Future Revision

During inference, we use predicted future information to revise an initial action before execution when its causal action gain falls below a revision threshold. For such actions, information about the difference between the action-conditioned future and the environmental future is used for a second action generation. The revised action is then combined with the initial action before execution.

Specifically, given the current observation $\mathbf{O}_t$ and task instruction $\mathbf{c}$, the policy first generates an initial action chunk $\mathbf{A}_t^{\text{init}}$. The PF-WM then predicts its action-conditioned future $\hat{\mathbf{O}}^{\text{init}}_{t+1:t+H}$ and the environmental future $\hat{\mathbf{O}}^{\text{env}}_{t+1:t+H}$. Using the same future evaluator as in training, we obtain Y_t^{init} and Y_t^{env} and compute the causal action gain $G_t^{\text{init}} = Y_t^{\text{init}} - Y_t^{\text{env}}$. When $G_t^{\text{init}} < \delta$, we perform an additional future-conditioned action generation, where δ denotes the revision threshold.

For actions that require revision, we use the internal representations of the two predicted futures to construct the feedback for the second generation. Let $\hat{\mathbf{Z}}^{\text{init}}_{t+1:t+H}$ and $\hat{\mathbf{Z}}^{\text{env}}_{t+1:t+H}$ denote the internal representations of the action-conditioned future and the environmental future, respectively. The former represents the predicted future under the initial action, while their difference captures the discrepancy between the action-conditioned and environmental futures. We construct the future

Table 1: Success rates (%) on the 24-task RoboCasa benchmark. The tasks are grouped into seven manipulation skill families: Pick&Place (8 tasks), Door (4 tasks), Drawer (2 tasks), Knob (2 tasks), Lever (3 tasks), Button (3 tasks), and Insertion (2 tasks). Avg. denotes the average success rate over all 24 tasks. Full results are provided in **Appendix I.1**.

Method	Pick&Place	Door	Drawer	Knob	Lever	Button	Insertion	Avg.
3D Diffuser Actor (Ke et al., 2024)	0.0	3.5	0.0	6.0	20.0	15.3	0.0	5.5
DP3 (Ze et al., 2024)	1.5	40.5	42.0	15.0	42.7	36.0	11.0	22.8
Diffusion Policy (ResNet) (Chi et al., 2025)	6.2	67.0	73.0	27.0	65.3	68.7	27.0	40.6
Diffusion Policy (CLIP) (Chi et al., 2025)	6.0	61.5	78.0	18.0	54.0	85.3	36.0	40.7
GR00T N1 (Bjorck et al., 2025)	22.6	58.0	89.0	41.5	63.0	78.0	48.0	49.7
FPV (Donat et al., 2025)	14.2	82.0	83.0	43.0	66.7	84.7	32.0	50.5
UVA (Li et al., 2025)	20.8	77.0	50.0	32.0	63.3	91.3	49.0	50.0
Video Policy (50 demos) (Liang et al., 2025)	50.5	90.5	71.0	18.0	67.3	91.3	49.0	63.2
Video Policy (300 demos) (Liang et al., 2025)	47.2	86.0	77.0	27.0	75.3	**97.3**	**64.0**	65.7
Cosmos Policy-Predict2.5 (Kim et al., 2026)	55.8	89.5	96.0	49.0	84.0	91.3	48.0	71.5
CAG-WAM (Ours)	**60.8**	**92.0**	**96.0**	**52.0**	**84.0**	92.0	51.0	**74.2**

Table 2: Success rates (%) on the four LIBERO task suites.

Method	Spatial	Object	Goal	Long	Avg.
Diffusion Policy (Chi et al., 2025)	78.3	92.5	68.3	50.5	72.4
Octo (Team et al., 2024)	78.9	85.7	84.6	51.1	75.1
OpenVLA (Kim et al., 2024)	84.7	88.4	79.2	53.7	76.5
TraceVLA (Zheng et al., 2025)	84.6	85.2	75.1	54.1	74.8
SpatialVLA (Qu et al., 2025)	88.2	89.9	78.6	55.5	78.1
UniVLA (Bu et al., 2025)	96.5	96.8	95.6	92.0	95.2
LingBot-VA (Li et al., 2026)	98.5	99.6	97.2	98.5	98.5
Cosmos Policy-Predict2 (Kim et al., 2026)	98.1	**100.0**	98.2	97.6	98.5
Cosmos Policy-Predict2.5 (Kim et al., 2026)	98.7	99.7	98.3	96.6	98.3
CAG-WAM (Ours)	**99.5**	**100.0**	**100.0**	**99.6**	**99.8**

feedback as $\hat{\mathbf{Z}}^{\mathrm{fb}}_{t+1:t+H} = \hat{\mathbf{Z}}^{\mathrm{init}}_{t+1:t+H} + \gamma(\hat{\mathbf{Z}}^{\mathrm{init}}_{t+1:t+H} - \hat{\mathbf{Z}}^{\mathrm{env}}_{t+1:t+H})$, where γ controls the contribution of the difference between the two future representations. The resulting future feedback is provided to the FC-AM together with the current observation and task instruction. The FC-AM then performs another complete generation to obtain a revised action chunk $\mathbf{A}^{\mathrm{rev}}_t$.

We then combine the revised action with the initial action before execution:

$$\mathbf{A}^{\mathrm{exec}}_t = \mathbf{A}^{\mathrm{init}}_t + \lambda(\mathbf{A}^{\mathrm{rev}}_t - \mathbf{A}^{\mathrm{init}}_t), \quad \lambda \in [0, 1], \tag{4}$$

where λ controls the contribution of the revised action to the final output.

Through this inference procedure, predicted future information is selectively incorporated into action generation according to the causal action gain. For actions that require revision, the difference between the action-conditioned and environmental future representations provides additional information for generating a revised action, while the final fusion controls how strongly the revised action modifies the initial policy output.

5 Experiments

In this section, we conduct experiments to evaluate the effectiveness of CAG-WAM. More details, experiments, and analysis are provided in **Appendix H–I** due to space limitations.

5.1 Experimental Setup

Benchmarks and Baselines We evaluate CAG-WAM on RoboCasa (Nasiriany et al., 2024) and LIBERO (Liu et al., 2023), using task success rate as the primary metric. The RoboCasa evaluation covers seven manipulation skill families, while LIBERO contains four suites covering spatial, object, goal, and long-horizon settings. We compare CAG-WAM with many representative methods, including 3D Diffuser Actor (Ke et al., 2024), DP3 (Ze et al., 2024), Diffusion Policy (Chi et al.,

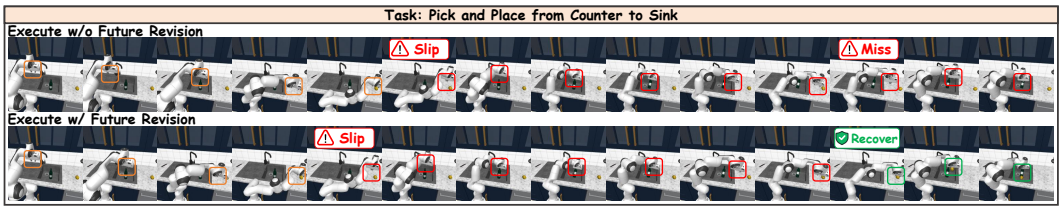


Figure 4: Visualization of CAG-WAM.

2025), GR00T N1 (Bjorck et al., 2025), FPV (Donat et al., 2025), Octo (Team et al., 2024), OpenVLA (Kim et al., 2024), TraceVLA (Zheng et al., 2025), SpatialVLA (Qu et al., 2025), UniVLA (Bu et al., 2025), UVA (Li et al., 2025), Video Policy (Liang et al., 2025), LingBot-VA (Li et al., 2026), Cosmos Policy-Predict2 (Kim et al., 2026), and Cosmos Policy-Predict2.5 (Kim et al., 2026).

Implementation Details We implement CAG-WAM based on publicly available robot policy and world-model codebases (Ali et al., 2025; Kim et al., 2026). During training, we use $K = 8$ candidate actions with learning rates of 5×10^{-5} for RoboCasa and 3×10^{-6} for LIBERO. During inference, we set $\delta = 0.05$, $\gamma = 1.0$, and $\lambda = 0.30$ for threshold, future feedback, and action fusion, respectively. More details are provided in **Appendix H**.

5.2 Main Results

Performance on RoboCasa **Table 1** reports the results on RoboCasa. CAG-WAM achieves an average success rate of 74.2% across the 24 RoboCasa tasks. These tasks span seven manipulation skill families: Pick&Place, Door, Drawer, Knob, Lever, Button, and Insertion, covering diverse manipulation settings. The task-level results in **Table 4** further show that the performance improvements are distributed across multiple individual tasks and skill categories. These results demonstrate the effectiveness of CAG-WAM across diverse RoboCasa manipulation settings.

Performance on Libero **Table 2** reports the results on the four LIBERO suites. CAG-WAM achieves an average success rate of 99.8%, with strong performance across all four task suites. It reaches 100% success on both Object and Goal, while maintaining high performance on Spatial and Long. These results support the effectiveness of CAG-WAM across manipulation settings involving spatial relations, object variations, task goals, and long-horizon execution.

5.3 Analysis

Visualization and Case Study We present a manipulation example in **Figure 4**. Without future revision, an intermediate slip leads to an inaccurate grasp and eventually causes the action execution to miss the target. With future revision, the policy can adjust the subsequent action after the slip and recover to complete the task. More results are provided in **Appendix I.3**.

6 Conclusion

In this work, we study action evaluation in World Action Models from a causal perspective. Our analysis considers how action-induced and environment-driven changes can jointly shape future evaluation, such that the overall future evaluation may not fully characterize the future change associated with a candidate action. To address this issue, we construct a structural causal model and propose CAG-WAM, which compares paired action-conditioned and environmental futures and defines their evaluation difference as causal action gain for the future change associated with the candidate action. During training, CAG is incorporated into GRPO to guide robot policy learning. During inference, CAG-WAM evaluates the initial action with CAG and, when needed, revises it using predicted future information. Extensive experiments support the effectiveness of CAG-WAM.

AI USE STATEMENT

In this work, we used generative AI tools for grammar and language polishing. They were not involved in research ideation, methodology, analysis, or substantive writing. The authors take full responsibility for all contents of the paper.

REFERENCES

Arslan Ali, Junjie Bai, Maciej Bala, Yogesh Balaji, Aaron Blakeman, Tiffany Cai, Jiaxin Cao, Tianshi Cao, Elizabeth Cha, Yu-Wei Chao, et al. World simulation with video foundation models for physical ai. *arXiv preprint arXiv:2511.00062*, 2025.

Ezra Ameperosa, Jeremy A Collins, Mrinal Jain, and Animesh Garg. Rocoda: Counterfactual data augmentation for data-efficient robot learning from demonstrations. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 13250–13256. IEEE, 2025.

Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

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

Lang Cao, Renhong Chen, Luyi Li, Peng Wang, Mofan Peng, and Yitong Li. Z-1: Efficient reinforcement learning for vision-language-action models. *arXiv preprint arXiv:2606.31846*, 2026.

Ronghan Chen, Yandan Yang, Zuojin Tang, Dongjie Huo, Tong Lin, Haoning Wu, Haoyun Liu, Yuzhi Chen, Lulu Zheng, Botai Yuan, et al. Abot-m0. 5: Unified mobility-and-manipulation world action model. *arXiv preprint arXiv:2607.00678*, 2026.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

Atalay Donat, Xiaogang Jia, Xi Huang, Aleksandar Taranovic, Denis Blessing, Ge Li, Hongyi Zhou, Hanyi Zhang, Rudolf Lioutikov, and Gerhard Neumann. Towards fusing point cloud and visual representations for imitation learning. *arXiv preprint arXiv:2502.12320*, 2025.

Chenghao Gu, Hanyang Yu, Jingbo Zhang, Haitao Lin, Wenyao Zhang, Jinghe Wang, Hanglei Jin, Shuzhao Xie, Jingyan Jiang, and Zhi Wang. Geniworld: A generalizable interactive world model for robotic manipulation via visual actions. *arXiv preprint arXiv:2608.06332*, 2026.

Miguel A Hernán and James M Robins. *Causal inference*. CRC Boca Raton, FL, 2010.

Yitao Jiang, Luyang Zhao, Muhao Chen, and Devin Balkcom. Flowmo-wm: A world model with object momentum and hidden ambient drift. *arXiv preprint arXiv:2606.13817*, 2026.

Tsung-Wei Ke, Nikolaos Gkanatsios, and Katerina Fragkiadaki. 3d diffuser actor: Policy diffusion with 3d scene representations. *arXiv preprint arXiv:2402.10885*, 2024.

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

Moo Jin Kim, Yihuai Gao, Tsung-Yi Lin, Yen-Chen Lin, Yunhao Ge, Grace Lam, Percy Liang, Shuran Song, Ming-Yu Liu, Chelsea Finn, et al. Cosmos policy: Fine-tuning video models for visuomotor control and planning. *arXiv preprint arXiv:2601.16163*, 2026.

Tabitha E Lee, Jialiang Alan Zhao, Amrita S Sawhney, Siddharth Girdhar, and Oliver Kroemer. Causal reasoning in simulation for structure and transfer learning of robot manipulation policies. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4776–4782. IEEE, 2021.

Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Fei Han, Mingrui Yu, Zelin Gao, Nan Xue, Xing Zhu, et al. Causal world modeling for robot control. *arXiv preprint arXiv:2601.21998*, 2026.

Shuang Li, Yihuai Gao, Dorsa Sadigh, and Shuran Song. Unified video action model, 2025. URL https://arxiv.org/abs/2503.00200.

Junbang Liang, Pavel Tokmakov, Ruoshi Liu, Sruthi Sudhakar, Paarth Shah, Rares Ambrus, and Carl Vondrick. Video generators are robot policies. *arXiv preprint arXiv:2508.00795*, 2025.

Yichao Liang, Dat Nguyen, Cambridge Yang, Tianyang Li, Joshua B Tenenbaum, Carl Edward Rasmussen, Adrian Weller, Zenna Tavares, Tom Silver, and Kevin Ellis. Exopredicator: Learning abstract models of dynamic worlds for robot planning. In *International Conference on Learning Representations*, volume 2026, pp. 49174–49214, 2026.

Wenjun Lin, Jensen Zhang, Kaitong Cai, and Keze Wang. Storm: Search-guided generative world models for robotic manipulation. *arXiv preprint arXiv:2512.18477*, 2025.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36:44776–44791, 2023.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl. *Advances in neural information processing systems*, 38:40783–40818, 2026a.

Yushan Liu, Peibo Sun, Shoujie Li, Yifan Xie, Lingfeng Zhang, Xintao Chao, Shiyuan Dong, Fang Chen, Xiao-Ping Zhang, and Wenbo Ding. Oa-wam: Object-addressable world action model for robust robot manipulation. *arXiv preprint arXiv:2605.06481*, 2026b.

Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. Robocasa: Large-scale simulation of everyday tasks for generalist robots. *arXiv preprint arXiv:2406.02523*, 2024.

Bikang Pan, Fan Liu, Haotao Lu, Jingya Wang, and Ye Shi. Selfwam: A self-grounded unified world action model for fast robot control. *arXiv preprint arXiv:2608.00725*, 2026.

Judea Pearl. *Causality*. Cambridge university press, 2009.

Quanquan Peng, Yutong Liang, Rui Yan, Nicklas Hansen, and Xiaolong Wang. Fact: Failure-aware causal training for world-action models. *arXiv preprint arXiv:2608.10232*, 2026.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. The MIT press, 2017.

Delin Qu, Haoming Song, Qizhi Chen, Yuanqi Yao, Xinyi Ye, Yan Ding, Zhigang Wang, JiaYuan Gu, Bin Zhao, Dong Wang, et al. Spatialvla: Exploring spatial representations for visual-language-action model. *arXiv preprint arXiv:2501.15830*, 2025.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

MotuBrain Team, Chendong Xiang, Fan Bao, Haitian Liu, Hengkai Tan, Hongzhe Bi, James Li, Jiabao Liu, Jingrui Pang, Kiro Jing, et al. Motubrain: An advanced world action model for robot control. *arXiv preprint arXiv:2604.27792*, 2026.

Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, et al. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*, 2024.

Fangyuan Wang, Ziyuan Wang, Guorui Pei, Mengshi Zhang, Canxi Liang, Jun Hu, Zhongxuan Li, Jinsong Wu, Ning Han, Zeqing Zhang, et al. World models for robotic manipulation: A survey. *arXiv preprint arXiv:2606.00113*, 2026a.

Kailin Wang, Haoxiang Jie, Yaoyuan Yan, Jiacheng Zhou, and Zhiyou Heng. Ea-wm: Event-aware world models with task-specification grounding for long-horizon manipulation. *arXiv preprint arXiv:2606.13053*, 2026b.

Chao Xue, Chaofan Zhang, Wenxuan Ma, Guocai Yao, Shaowei Cui, and Shuo Wang. Hitac-wam: A hierarchical tactile world action model for contact-rich robot manipulation. *arXiv preprint arXiv:2608.19574*, 2026.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, et al. World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*, 2026.

Yanjie Ze, Gu Zhang, Kangning Zhang, Chenyuan Hu, Muhan Wang, and Huazhe Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. *arXiv preprint arXiv:2403.03954*, 2024.

Yan Zhang, Yinan Wu, Haoran Duan, and Jungong Han. Cofactvla: Deconfounding vision-language-action models via counterfactual intervention. *arXiv preprint arXiv:2608.04396*, 2026.

Baining Zhao, Ziyou Wang, Jianjie Fang, Chen Gao, Fanhang Man, Jinqiang Cui, Xin Wang, Xinlei Chen, Yong Li, and Wenwu Zhu. Embodied-r: Collaborative framework for activating embodied spatial reasoning in foundation models via reinforcement learning. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 11071–11080, 2025.

Ruijie Zheng, Yongyuan Liang, Shuaiyi Huang, Jianfeng Gao, Hal Daumé III, Andrey Kolobov, Furong Huang, and Jianwei Yang. Tracevla: Visual trace prompting enhances spatial-temporal awareness for generalist robotic policies. In *International Conference on Learning Representations*, volume 2025, pp. 54277–54296, 2025.

Pengfei Zhou, Shengcong Chen, Di Chen, Jiaxu Wang, Rongjun Jin, Bingwen Zhu, Yike Pan, Songen Gu, Kuanning Wang, Shufeng Nan, et al. τ_0-wm: A unified video-action world model for robotic manipulation. *arXiv preprint arXiv:2606.01027*, 2026.

Jian Zhu, Jianjun Zhang, Taiyi Su, Tianbin Liu, Zhangyuan Wang, Kai Xie, Zitai Huang, Chong Ma, Youzhang He, Tianjian Wang, et al. Dswam: A dual-system world action foundation model for fine-grained robot manipulation. *arXiv preprint arXiv:2607.04927*, 2026.

APPENDIX

The appendix is organized into several sections:

- **Appendix A** provides details for all notations used in this paper.
- **Appendix B** provides the preliminaries used in this paper.
- **Appendix C** provides theoretical analysis and additional justification.
- **Appendix D** provides the pseudo-code of our method.
- **Appendix E** provides details for the benchmarks used in the experiments.
- **Appendix F** provides details for the baselines mentioned in the main text.
- **Appendix G** provides more discussion about the proposed methodology.
- **Appendix H** contains details for the implementation of the experiment.
- **Appendix I** provides the full results and analyses of the experiment.
- **Appendix J** illustrates the broader impacts and limitations of this paper.

A NOTATION

In this section, we briefly summarize the main symbols used throughout the paper. We organize the notation according to its role in CAG-WAM.

- **Notations of Observations, Actions, and Future Prediction**
  - $\mathbf{O}_t$: the current robot observation at decision time t, including visual and proprioceptive observations.
  - $\mathbf{c}$: the task instruction.
  - π_θ: the action policy parameterized by θ.
  - θ: the parameters of the current action policy.
  - K: the number of candidate action chunks generated for the current observation.
  - $\mathbf{A}_t^i$: the i-th candidate action chunk generated at decision time t, where $i \in \{1, \dots, K\}$.
  - $\mathbf{A}_t^i = [\mathbf{a}_t^i, \dots, \mathbf{a}_{t+L-1}^i]$: the sequence of low-level robot actions contained in candidate action chunk $\mathbf{A}_t^i$.
  - $\mathbf{a}_{t+\ell}^i$: an individual low-level robot action in the i-th candidate action chunk.
  - L: the length of an action chunk.
  - p_ϕ: the learned predictive world model parameterized by ϕ.
  - ϕ: the parameters of the predictive world model.
  - H: the future-prediction horizon.
  - $\hat{\mathbf{O}}_{t+1:t+H}^i$: the predicted future observation sequence under candidate action $\mathbf{A}_t^i$.
  - $\mathcal{V}$: the future evaluator used to evaluate predicted futures.
  - $Y_t^i = \mathcal{V}(\hat{\mathbf{O}}_{t+1:t+H}^i, \mathbf{c})$: the realized future evaluation associated with candidate action $\mathbf{A}_t^i$.
- **Notations of Structural Causal Analysis**
  - $\mathbf{A}_t$: the candidate-action variable in the structural causal model. At the decision level, it represents the entire candidate action chunk as a single intervention variable.
  - $\mathbf{U}_t$: exogenous environmental factors that are not directly controlled by the current candidate action and may affect future evolution over the prediction horizon.
  - $\mathbf{E}_{t+1}$: environment-driven changes whose evolution is not causally induced by the current candidate action.
  - $\mathbf{M}_{t+1}$: action-induced changes causally downstream of the current candidate action.
  - $\mathbf{O}_{t+1}$: the future-outcome variable in the SCM jointly affected by $\mathbf{E}_{t+1}$ and $\mathbf{M}_{t+1}$. It compactly represents the future outcome over the prediction horizon rather than only a single low-level transition.

- Y_t: the future evaluation in the structural causal model.
- $\mathbf{a}$: a possible realization of the candidate-action variable $\mathbf{A}_t$.
- $\mathbf{o}$: a possible realization of the current observation $\mathbf{O}_t$.
- $p(\mathbf{o} \mid \mathbf{A}_t = \mathbf{a})$: the observation distribution conditioned on the candidate action.
- $p(\mathbf{o})$: the marginal observation distribution used in backdoor adjustment.
- $do(\mathbf{A}_t = \mathbf{a})$: the intervention that fixes the candidate-action variable to $\mathbf{a}$ and removes its incoming dependence on the current observation.
- $\mathbb{E}[Y_t \mid \mathbf{A}_t = \mathbf{a}]$: the observational expected future evaluation associated with action $\mathbf{a}$.
- $\mathbb{E}[Y_t \mid do(\mathbf{A}_t = \mathbf{a})]$: the marginal interventional future evaluation obtained through backdoor adjustment.
- $\mathbb{E}[Y_t \mid do(\mathbf{A}_t = \mathbf{a}), \mathbf{O}_t = \mathbf{o}]$: the observation-conditioned interventional future evaluation at a realized current observation.
- $\mathbf{a}^{\mathrm{ref}}$: the comparison action condition used to define the reference-relative causal contrast. It is another intervention value of the same action variable $\mathbf{A}_t$.
- $\tau_t(\mathbf{a}, \mathbf{a}^{\mathrm{ref}} \mid \mathbf{o})$: the reference-relative conditional average causal contrast, defined as

$$\begin{aligned}\tau_t(\mathbf{a}, \mathbf{a}^{\mathrm{ref}} \mid \mathbf{o}) = \mathbb{E}\left[Y_t \mid do(\mathbf{A}_t = \mathbf{a}), \mathbf{O}_t = \mathbf{o}\right] \\ - \mathbb{E}\left[Y_t \mid do(\mathbf{A}_t = \mathbf{a}^{\mathrm{ref}}), \mathbf{O}_t = \mathbf{o}\right].\end{aligned}$$

- **Notations of Causal Action Gain**
  - $\hat{\mathbf{O}}^{\mathrm{env}}_{t+1:t+H}$: the environmental future predicted from the current observation and task instruction under the common comparison action condition. In our implementation, this condition is HOLD.
  - $Y_t^{\mathrm{env}} = \mathcal{V}(\hat{\mathbf{O}}^{\mathrm{env}}_{t+1:t+H}, \mathbf{c})$: the realized evaluation of the environmental future.
  - $G_t^i = Y_t^i - Y_t^{\mathrm{env}}$: the realized causal action gain of candidate action $\mathbf{A}_t^i$. It is a contrastive quantity defined relative to the fixed comparison action condition rather than an intrinsic property of the candidate action alone. For readability, its dependence on the comparison condition, current observation, and task instruction is suppressed in the compact notation G_t^i.
  - $G_t^i > 0$: the candidate-conditioned future receives a higher realized evaluation than the environmental future.
  - $G_t^i < 0$: the candidate-conditioned future receives a lower realized evaluation than the environmental future.
  - $\bar{Y}_{\phi,t}(\mathbf{a} \mid \mathbf{o}, \mathbf{c})$: the model-level expected future evaluation under action condition $\mathbf{a}$.
  - $\bar{G}_{\phi,t}(\mathbf{a}, \mathbf{a}^{\mathrm{ref}} \mid \mathbf{o}, \mathbf{c})$: the model-level expected difference between the candidate and comparison action conditions.
  - $\varepsilon_{\phi,t}(\mathbf{a} \mid \mathbf{o}, \mathbf{c})$: the combined future-prediction and future-evaluation approximation error for action condition $\mathbf{a}$.
  - $\Delta^i_{t,\mathrm{CAG}}$: the policy-optimization advantage assigned to candidate action $\mathbf{A}_t^i$, defined as $\Delta^i_{t,\mathrm{CAG}} = G_t^i$.
- **Notations of CAG-Guided Policy Optimization**
  - T: the number of generation steps used by the flow-based action policy.
  - $\tau \in \{1, \ldots, T\}$: the index of a generation step.
  - $\mathbf{X}^i_{t,\tau}$: the intermediate flow variable of candidate action i at generation step τ.
  - $\pi_{\theta_{\mathrm{old}}}$: the sampling policy used to generate candidate actions before the current policy update.
  - π_{ref}: the reference policy used for KL regularization.
  - $\rho^i_{t,\tau}(\theta)$: the transition-level importance ratio between the current policy and the sampling policy for candidate i at generation step τ.
  - ϵ: the clipping coefficient in the GRPO objective.
  - β: the coefficient of the KL regularization term.
  - $D_{\mathrm{KL}}(\pi_\theta \| \pi_{\mathrm{ref}})$: the KL divergence between the current policy and the reference policy.

- $\mathcal{J}_{\mathrm{CAG}}(\theta)$: the clipped GRPO objective that uses causal action gain as the optimization advantage.

- **Notations of Inference-Time Action Revision**
  - $\mathbf{A}_t^{\mathrm{init}}$: the initial action chunk generated from the current observation and task instruction.
  - $\hat{\mathbf{O}}_{t+1:t+H}^{\mathrm{init}}$: the action-conditioned future predicted for the initial action.
  - Y_t^{init}: the evaluation of the action-conditioned future corresponding to the initial action.
  - $G_t^{\mathrm{init}} = Y_t^{\mathrm{init}} - Y_t^{\mathrm{env}}$: the realized causal action gain of the initial action.
  - δ: the revision threshold used to determine when Future Revision is activated, according to $G_t^{\mathrm{init}} < \delta$.
  - $\hat{\mathbf{Z}}_{t+1:t+H}^{\mathrm{init}}$: the internal representation of the action-conditioned future corresponding to the initial action.
  - $\hat{\mathbf{Z}}_{t+1:t+H}^{\mathrm{env}}$: the internal representation of the environmental future.
  - $\mathbf{D}_t = \hat{\mathbf{Z}}_{t+1:t+H}^{\mathrm{init}} - \hat{\mathbf{Z}}_{t+1:t+H}^{\mathrm{env}}$: the representation-level discrepancy between the action-conditioned and environmental futures.
  - $\hat{\mathbf{Z}}_{t+1:t+H}^{\mathrm{fb}}$: the future-feedback representation used as additional conditioning information for the second action generation.
  - γ: the coefficient controlling the contribution of the paired-future discrepancy to the future-feedback representation.
  - $\mathbf{A}_t^{\mathrm{rev}}$: the revised action chunk produced by the additional future-conditioned generation.
  - $\lambda \in [0, 1]$: the coefficient controlling the contribution of the revised action to the executed action.
  - $\mathbf{A}_t^{\mathrm{exec}}$: the final action chunk executed by the robot after combining the initial and revised actions.

B PRELIMINARIES

B.1 INTERVENTIONS, BACKDOOR PATHS, AND BACKDOOR ADJUSTMENT

We briefly introduce the standard concepts of intervention, backdoor paths, and backdoor adjustment in a directed acyclic graph (DAG) (Pearl, 2009).

Definition (Path blocking and d-separation)**.** Consider a DAG $\mathcal{G}$ and a path between two variables X and Y. A node on the path is a *collider* if the two adjacent edges on the path both point into that node, e.g., $X \to Z \leftarrow Y$. Given a set of variables $\mathbf{Z}$, a path is blocked by $\mathbf{Z}$ if either (i) the path contains a non-collider in $\mathbf{Z}$, or (ii) the path contains a collider such that neither the collider nor any of its descendants belongs to $\mathbf{Z}$. Two variables are d-separated by $\mathbf{Z}$ if every path between them is blocked by $\mathbf{Z}$.

Definition (Intervention)**.** In a structural causal model, an intervention $do(X = x)$ sets the variable X to x by replacing its structural assignment with the constant x. Graphically, this operation removes the incoming edges into X while leaving the remaining structural mechanisms unchanged. The resulting distribution is called an interventional distribution.

Definition (Backdoor path)**.** A *backdoor path* from X to Y is a path between X and Y whose first edge points into X. An open backdoor path can transmit non-causal association between X and Y.

Definition (Backdoor criterion)**.** A set of variables $\mathbf{Z}$ satisfies the backdoor criterion relative to an ordered pair (X, Y) if (i) no variable in $\mathbf{Z}$ is a descendant of X, and (ii) $\mathbf{Z}$ blocks every path between X and Y that contains an arrow into X. When these conditions hold, conditioning on $\mathbf{Z}$ blocks the associations transmitted through the backdoor paths while leaving the directed causal paths from X to Y unblocked.

Proposition (Backdoor adjustment)**.** *If $\mathbf{Z}$ satisfies the backdoor criterion relative to (X, Y), the interventional distribution of Y under $do(X = x)$ is identifiable from the observational distribution as*

$$p(y \mid do(X = x)) = \int p(y \mid X = x, \mathbf{Z} = \mathbf{z})\, p(\mathbf{z})\, \mathrm{d}\mathbf{z}. \tag{5}$$

For a scalar outcome Y, the corresponding interventional expectation is

$$\mathbb{E}[Y \mid do(X = x)] = \int \mathbb{E}[Y \mid X = x, \mathbf{Z} = \mathbf{z}]\, p(\mathbf{z})\, \mathrm{d}\mathbf{z}. \tag{6}$$

When $\mathbf{Z}$ is discrete, the integrals are replaced by sums.

The distinction from the observational conditional expectation is that

$$\mathbb{E}[Y \mid X = x] = \int \mathbb{E}[Y \mid X = x, \mathbf{Z} = \mathbf{z}]\, p(\mathbf{z} \mid X = x)\, \mathrm{d}\mathbf{z}, \tag{7}$$

whereas backdoor adjustment averages the conditional outcome under the common marginal distribution $p(\mathbf{z})$.

B.2 Group Relative Policy Optimization

Group Relative Policy Optimization (GRPO) is a policy optimization method that estimates relative advantages from a group of sampled outputs, avoiding the need for a separately learned value function (Shao et al., 2024). Given a context $\mathbf{x}$, let $\{\mathbf{y}^i\}_{i=1}^G$ denote a group of G outputs sampled from the old policy $\pi_{\theta_{\text{old}}}(\cdot \mid \mathbf{x})$, and let R^i denote the scalar reward assigned to the i-th output. The group mean and standard deviation are

$$\bar{R} = \frac{1}{G}\sum_{i=1}^{G} R^i, \qquad \sigma_R = \sqrt{\frac{1}{G}\sum_{i=1}^{G}(R^i - \bar{R})^2}. \tag{8}$$

GRPO constructs a group-relative advantage by normalizing each reward with respect to the sampled group:

$$\widehat{A}^i = \frac{R^i - \bar{R}}{\sigma_R}. \tag{9}$$

Thus, the advantage reflects the relative quality of an output among the samples generated for the same context.

To update the policy, GRPO follows a PPO-style clipped surrogate objective. For the i-th sampled output, define the importance ratio

$$\rho^i(\theta) = \frac{\pi_\theta(\mathbf{y}^i \mid \mathbf{x})}{\pi_{\theta_{\text{old}}}(\mathbf{y}^i \mid \mathbf{x})}. \tag{10}$$

The corresponding clipped objective can be written as

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}\left[\frac{1}{G}\sum_{i=1}^{G} \min\left(\rho^i(\theta)\widehat{A}^i, \text{clip}\left(\rho^i(\theta), 1-\epsilon, 1+\epsilon\right)\widehat{A}^i\right) - \beta D_{\text{KL}}\left(\pi_\theta \| \pi_{\text{ref}}\right)\right], \tag{11}$$

where ϵ is the clipping coefficient, π_{ref} is a reference policy, and β controls the strength of KL regularization. The clipping operation limits excessively large policy updates, while the KL term regularizes the optimized policy toward the reference policy.

B.3 Flow-GRPO for Flow-Matching Policies

Flow-matching policies generate samples through a continuous deterministic flow. We use $s \in [0, 1]$ to denote the continuous flow time and distinguish it from the discrete generation steps introduced below. The generation process is described by the ordinary differential equation

$$\mathrm{d}\mathbf{X}_s = \mathbf{v}_\theta(\mathbf{X}_s, s; \mathbf{x})\, \mathrm{d}s, \tag{12}$$

where $\mathbf{x}$ denotes the conditioning context and $\mathbf{v}_\theta$ is the learned velocity field. Unlike a stochastic policy, this deterministic formulation does not directly provide tractable transition likelihoods for importance-ratio-based policy optimization.

Flow-GRPO (Liu et al., 2026a) addresses this issue by introducing a stochastic formulation whose marginal density evolution is consistent with that of the original flow process. For rectified flow, the interpolation between a data sample $\mathbf{X}_0$ and Gaussian noise $\boldsymbol{\xi}$ is

$$\mathbf{X}_s = (1 - s)\mathbf{X}_0 + s\boldsymbol{\xi}, \qquad \boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{13}$$

and the corresponding reverse-time stochastic sampler used by Flow-GRPO can be written as

$$\mathrm{d}\mathbf{X}_s = \left[\mathbf{v}_\theta(\mathbf{X}_s, s; \mathbf{x}) + \frac{\sigma_s^2}{2s}\left(\mathbf{X}_s + (1-s)\mathbf{v}_\theta(\mathbf{X}_s, s; \mathbf{x})\right)\right]\mathrm{d}s + \sigma_s\,\mathrm{d}\overline{\mathbf{W}}_s, \tag{14}$$

where $\overline{\mathbf{W}}_s$ denotes the reverse-time Wiener process and σ_s specifies the stochastic schedule. We refer to Liu et al. (2026a) for the derivation of this stochastic formulation and its relation to the deterministic flow.

Discretizing **Eq. 14** with Euler–Maruyama yields a stochastic transition at each generation step. Let $\mathbf{X}_\tau$ denote the intermediate variable at generation step τ, and let $h_\tau > 0$ denote the magnitude of the corresponding flow-time interval. The transition can be written as

$$\mathbf{X}_{\tau-1} = \boldsymbol{\mu}_{\theta,\tau} + \sigma_{s_\tau}\sqrt{h_\tau}\,\boldsymbol{\xi}_\tau, \qquad \boldsymbol{\xi}_\tau \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{15}$$

where $\boldsymbol{\mu}_{\theta,\tau}$ denotes the deterministic part of the discretized update. Consequently, each generation step admits the Gaussian transition likelihood

$$p_\theta\left(\mathbf{X}_{\tau-1} \mid \mathbf{X}_\tau, \mathbf{x}\right) = \mathcal{N}\left(\mathbf{X}_{\tau-1}; \boldsymbol{\mu}_{\theta,\tau}, \sigma_{s_\tau}^2 h_\tau \mathbf{I}\right). \tag{16}$$

For a generation trajectory $\mathcal{X} = \{\mathbf{X}_T, \ldots, \mathbf{X}_0\}$, the transition formulation gives

$$P_\theta\left(\mathcal{X} \mid \mathbf{x}\right) = p(\mathbf{X}_T)\prod_{\tau=1}^{T} p_\theta\left(\mathbf{X}_{\tau-1} \mid \mathbf{X}_\tau, \mathbf{x}\right), \tag{17}$$

where the initial noise distribution $p(\mathbf{X}_T)$ does not depend on θ. Therefore,

$$\nabla_\theta \log P_\theta\left(\mathcal{X} \mid \mathbf{x}\right) = \sum_{\tau=1}^{T} \nabla_\theta \log p_\theta\left(\mathbf{X}_{\tau-1} \mid \mathbf{X}_\tau, \mathbf{x}\right). \tag{18}$$

Using the same stochastic schedule for the current and sampling policies, the transition-level importance ratio is

$$\rho_\tau(\theta) = \frac{p_\theta\left(\mathbf{X}_{\tau-1} \mid \mathbf{X}_\tau, \mathbf{x}\right)}{p_{\theta_{\mathrm{old}}}\left(\mathbf{X}_{\tau-1} \mid \mathbf{X}_\tau, \mathbf{x}\right)}. \tag{19}$$

Because the two transition distributions share the same covariance, the corresponding log-ratio is

$$\log \rho_\tau(\theta) = \frac{\|\mathbf{X}_{\tau-1} - \boldsymbol{\mu}_{\theta_{\mathrm{old}},\tau}\|_2^2 - \|\mathbf{X}_{\tau-1} - \boldsymbol{\mu}_{\theta,\tau}\|_2^2}{2\sigma_{s_\tau}^2 h_\tau}. \tag{20}$$

Similarly, if the current and reference policies use the same transition covariance, their conditional KL divergence has the closed form

$$D_{\mathrm{KL}}\left(p_\theta \| p_{\mathrm{ref}}\right) = \frac{\|\boldsymbol{\mu}_{\theta,\tau} - \boldsymbol{\mu}_{\mathrm{ref},\tau}\|_2^2}{2\sigma_{s_\tau}^2 h_\tau}. \tag{21}$$

These transition-level likelihood quantities allow the clipped GRPO objective introduced in **Appendix B.2** to be applied to flow-matching policies.

B.4 Score-Function Policy Gradients and Action-Independent Baselines

Consider a stochastic policy $\pi_\theta(\mathbf{a} \mid \mathbf{x})$ and an objective

$$\mathcal{J}(\theta) = \mathbb{E}_{\mathbf{a}\sim\pi_\theta(\cdot|\mathbf{x})}\left[R(\mathbf{a})\right], \tag{22}$$

where $R(\mathbf{a})$ is treated as fixed with respect to θ during the policy update. Using the score-function identity,

$$\nabla_\theta \mathcal{J}(\theta) = \mathbb{E}\left[R(\mathbf{a})\nabla_\theta \log \pi_\theta(\mathbf{a} \mid \mathbf{x})\right]. \tag{23}$$

For any baseline $b(\mathbf{x})$ that does not depend on the sampled action,

$$\begin{aligned}\mathbb{E}\left[b(\mathbf{x})\nabla_\theta \log \pi_\theta(\mathbf{a} \mid \mathbf{x})\right] &= b(\mathbf{x})\nabla_\theta \int \pi_\theta(\mathbf{a} \mid \mathbf{x})\,\mathrm{d}\mathbf{a} \\ &= 0.\end{aligned} \tag{24}$$

Therefore, subtracting an action-independent baseline does not change the expected score-function policy gradient.

C Theoretical Analysis

C.1 Causal Interpretation and Backdoor Adjustment

Backdoor-adjusted action evaluation. Under the structural causal model in **Section 3**, the candidate action $\mathbf{A}_t$ and the future evaluation Y_t are connected by the backdoor path $\mathbf{A}_t \leftarrow \mathbf{O}_t \rightarrow \mathbf{E}_{t+1} \rightarrow \mathbf{O}_{t+1} \rightarrow Y_t$. For clarity, we condition on the task instruction $\mathbf{c}$ throughout this analysis and omit it from the causal notation. For action values within the relevant support, the future evaluation associated with an observed action can be written as

$$\mathbb{E}\left[Y_t \mid \mathbf{A}_t = \mathbf{a}\right] = \int \mathbb{E}\left[Y_t \mid \mathbf{A}_t = \mathbf{a}, \mathbf{O}_t = \mathbf{o}\right] p\left(\mathbf{o} \mid \mathbf{A}_t = \mathbf{a}\right) \, \mathrm{d}\mathbf{o}. \tag{25}$$

Because the candidate action is generated conditioned on $\mathbf{O}_t$, the weighting distribution $p(\mathbf{o} \mid \mathbf{A}_t = \mathbf{a})$ can depend on the selected action. Meanwhile, $\mathbf{O}_t$ also affects the future evaluation through the environment-driven changes. Therefore, the association in **Eq. 25** may contain influence transmitted through the identified backdoor path.

Intervening on the action variable removes the incoming dependence $\mathbf{O}_t \rightarrow \mathbf{A}_t$ while leaving the remaining causal mechanisms unchanged. Using the law of total expectation, the corresponding interventional evaluation can be written as

$$\mathbb{E}\left[Y_t \mid do(\mathbf{A}_t = \mathbf{a})\right] = \int \mathbb{E}\left[Y_t \mid do(\mathbf{A}_t = \mathbf{a}), \mathbf{O}_t = \mathbf{o}\right] p\left(\mathbf{o} \mid do(\mathbf{A}_t = \mathbf{a})\right) \, \mathrm{d}\mathbf{o}. \tag{26}$$

Since $\mathbf{O}_t$ is determined before the current action, intervening on $\mathbf{A}_t$ does not change its distribution, giving

$$p\left(\mathbf{o} \mid do(\mathbf{A}_t = \mathbf{a})\right) = p(\mathbf{o}). \tag{27}$$

Moreover, under the stated SCM and the identification assumptions specified below, conditioning on $\mathbf{O}_t$ blocks the identified backdoor path, so that

$$\mathbb{E}\left[Y_t \mid do(\mathbf{A}_t = \mathbf{a}), \mathbf{O}_t = \mathbf{o}\right] = \mathbb{E}\left[Y_t \mid \mathbf{A}_t = \mathbf{a}, \mathbf{O}_t = \mathbf{o}\right]. \tag{28}$$

Substituting **Eq. 27** and **Eq. 28** into **Eq. 26** gives

$$\mathbb{E}\left[Y_t \mid do(\mathbf{A}_t = \mathbf{a})\right] = \int \mathbb{E}\left[Y_t \mid \mathbf{A}_t = \mathbf{a}, \mathbf{O}_t = \mathbf{o}\right] p(\mathbf{o}) \, \mathrm{d}\mathbf{o}. \tag{29}$$

Compared with **Eq. 25**, the action-dependent weighting $p(\mathbf{o} \mid \mathbf{A}_t = \mathbf{a})$ is replaced by the common observation distribution $p(\mathbf{o})$. Equation 29 is therefore a marginal interventional evaluation obtained by averaging observation-conditioned evaluations over a common observation distribution. The adjustment accounts for the association transmitted through the identified backdoor path. Importantly, it does not remove environmental evolution itself: the current observation and exogenous environmental factors can still affect the future through the environment-driven changes after the intervention.

Observation-conditioned interventional evaluation. In robot action generation, the current observation has already been realized before the action is selected. For a fixed observation $\mathbf{O}_t = \mathbf{o}$, **Eq. 28** gives

$$\mathbb{E}\left[Y_t \mid do(\mathbf{A}_t = \mathbf{a}), \mathbf{O}_t = \mathbf{o}\right] = \mathbb{E}\left[Y_t \mid \mathbf{A}_t = \mathbf{a}, \mathbf{O}_t = \mathbf{o}\right]. \tag{30}$$

This observation-conditioned evaluation and the marginal interventional evaluation in **Eq. 29** are related but are not the same quantity. The former evaluates an action at a realized current observation, whereas the latter averages such conditional evaluations over the observation distribution. Accordingly, CAG-WAM does not explicitly estimate the integral in **Eq. 29**. Instead, action generation and policy optimization operate at the realized current observation.

Identification, support, and model-approximation assumptions. The identification in **Eq. 30** follows from the causal structure represented by the SCM rather than from the learned world model itself. We assume that, conditioned on the current observation $\mathbf{O}_t$ and task instruction $\mathbf{c}$, there is no remaining unblocked backdoor path between the current action $\mathbf{A}_t$ and the future outcome. Equivalently, the conditioning information is assumed to contain the variables required by the adjustment

set represented in the SCM. If additional latent state or interaction history creates a common cause of the current action and future outcome beyond this conditioning information, the same identification would require that information to be included in the adjustment set.

We also assume that both the candidate action and the specified comparison action condition lie within the relevant conditional action support represented by the learned world model. Because the robot actions are continuous, this support requirement refers to the support of the conditional action density rather than positive probability at an individual action value. Predictions outside this support should instead be regarded as model extrapolation.

Finally, within this support, the learned world model is assumed to approximate the corresponding action-conditioned observational distribution. Restoring the task instruction $\mathbf{c}$ explicitly, these two steps can be summarized as

$$\begin{aligned} & p_\phi\left(\mathbf{O}_{t+1:t+H} \mid \mathbf{O}_t=\mathbf{o}, \mathbf{A}_t=\mathbf{a}, \mathbf{c}\right) \\ & \quad \approx p\left(\mathbf{O}_{t+1:t+H} \mid \mathbf{A}_t=\mathbf{a}, \mathbf{O}_t=\mathbf{o}, \mathbf{c}\right) \\ & \quad = p\left(\mathbf{O}_{t+1:t+H} \mid do(\mathbf{A}_t=\mathbf{a}), \mathbf{O}_t=\mathbf{o}, \mathbf{c}\right). \end{aligned} \tag{31}$$

The first relation in **Eq. 31** is a model-approximation assumption. The second equality follows from the causal identification implied by the SCM and the corresponding backdoor adjustment. Thus, the learned conditional world model does not by itself establish a causal interpretation; it provides a practical approximation to the identified interventional distribution under the stated assumptions.

Reference-relative conditional average causal contrast. Once the observation-conditioned interventional evaluation in **Eq. 30** is identified, the causal outcome of a candidate action can be compared with that of a specified comparison action condition under the same realized observation. Let $\mathbf{a}^{\mathrm{ref}}$ denote this common comparison action condition. Here, $\mathbf{a}^{\mathrm{ref}}$ denotes another intervention value of the same action variable $\mathbf{A}_t$, rather than an additional variable in the SCM. We define the reference-relative conditional average causal contrast as

$$\begin{aligned} \tau_t\left(\mathbf{a}, \mathbf{a}^{\mathrm{ref}} \mid \mathbf{o}\right) = & \mathbb{E}\left[Y_t \mid do(\mathbf{A}_t=\mathbf{a}), \mathbf{O}_t=\mathbf{o}\right] \\ & - \mathbb{E}\left[Y_t \mid do(\mathbf{A}_t=\mathbf{a}^{\mathrm{ref}}), \mathbf{O}_t=\mathbf{o}\right]. \end{aligned} \tag{32}$$

This estimand measures the difference between the two expected interventional future evaluations at the same realized current observation.

The quantity in **Eq. 32** is an expectation-level causal contrast. It should be distinguished from a unit-level counterfactual contrast that compares two potential outcomes while holding a particular realization of the exogenous variables fixed across the interventions. CAG-WAM does not target or identify such a unit-level counterfactual quantity. Instead, its causal target is the difference between the observation-conditioned interventional expectations in **Eq. 32**.

Under the assumptions above, applying **Eq. 28** to both terms in **Eq. 32** gives

$$\begin{aligned} \tau_t\left(\mathbf{a}, \mathbf{a}^{\mathrm{ref}} \mid \mathbf{o}\right) = & \mathbb{E}\left[Y_t \mid \mathbf{A}_t=\mathbf{a}, \mathbf{O}_t=\mathbf{o}\right] \\ & - \mathbb{E}\left[Y_t \mid \mathbf{A}_t=\mathbf{a}^{\mathrm{ref}}, \mathbf{O}_t=\mathbf{o}\right]. \end{aligned} \tag{33}$$

Thus, once a comparison action condition is specified, the SCM and backdoor adjustment identify the two observation-conditioned interventional expectations whose difference defines the reference-relative conditional average causal contrast. The choice of the comparison action condition is separate from the backdoor adjustment itself: the causal analysis identifies the intervention-specific expectations, while the comparison condition specifies the contrast of interest.

Relation to paired-future causal action gain. CAG-WAM uses model-predicted futures to approximate the expectation-level contrast in **Eq. 32**. For a realized current observation $\mathbf{O}_t=\mathbf{o}$, let $\hat{\mathbf{O}}^i_{t+1:t+H}$ denote a future predicted under candidate action $\mathbf{A}^i_t$, and let $\hat{\mathbf{O}}^{\mathrm{env}}_{t+1:t+H}$ denote the environmental future predicted under the common comparison action condition $\mathbf{a}^{\mathrm{ref}}$. Both futures are generated from the same current observation and task instruction. Their realized evaluations define the causal action gain as

$$\begin{aligned} G^i_t &= Y^i_t - Y^{\mathrm{env}}_t \\ &= V\left(\hat{\mathbf{O}}^i_{t+1:t+H}, \mathbf{c}\right) - V\left(\hat{\mathbf{O}}^{\mathrm{env}}_{t+1:t+H}, \mathbf{c}\right). \end{aligned} \tag{34}$$

Causal action gain is therefore a contrastive quantity defined relative to the specified comparison action condition, rather than an intrinsic causal property of the candidate action alone. Changing the comparison action condition can in general change the magnitude and sign of the resulting gain. For readability, we retain the compact notation G_t^i throughout the paper and suppress its explicit dependence on $\mathbf{a}^{\mathrm{ref}}$, $\mathbf{O}_t$, and $\mathbf{c}$, which are fixed within each paired-future comparison.

Because the two predicted futures are not required to share the same stochastic realization, an individual value of G_t^i in **Eq. 34** should not be interpreted as a unit-level counterfactual difference under fixed exogenous randomness. Rather, it is a realized stochastic model-based comparison between the candidate and comparison action conditions. Taking expectation over the stochasticity of the learned future generation yields the corresponding model-level expected contrast.

Under the identification, support, and model-approximation assumptions stated above, and assuming that the future evaluator accurately approximates the corresponding future evaluations, this model-level expected contrast provides a model-based approximation to the reference-relative conditional average causal contrast in **Eq. 32**. Equivalently,

$$\mathbb{E}_\phi \left[G_t^i \mid \mathbf{O}_t = \mathbf{o}, \mathbf{c}, \mathbf{A}_t^i = \mathbf{a} \right] \approx \tau_t \left(\mathbf{a}, \mathbf{a}^{\mathrm{ref}} \mid \mathbf{o} \right), \tag{35}$$

where the expectation on the left is taken over the stochasticity of the learned future generation. The distinction between this model-level expected contrast and the realized sampled causal action gain is discussed further in **Appendix C.2**.

Under the assumptions above, backdoor adjustment identifies the observation-conditioned interventional expectations that define the causal contrast, while the learned world model and future evaluator approximate their outcomes. The realized causal action gain is a stochastic model-based comparison, whose model-level expectation approximates this target. The target is expectation-level and comparison-dependent, rather than a unit-level counterfactual effect or an intrinsic causal property of the candidate action.

After conditioning on $\mathbf{O}_t$, subtracting the environmental-future evaluation does not perform an additional deconfounding step. It instead provides the comparison term in the causal contrast defined in **Eq. 32**. The environmental future need not represent a pure environment-only trajectory or exactly isolate the environment-driven component of the SCM. Its operational definition is given in **Appendix G.5**.

C.2 Properties of Paired Future Evaluation

Meaning of paired future evaluation. For a fixed current observation $\mathbf{O}_t = \mathbf{o}$ and task instruction $\mathbf{c}$, CAG-WAM compares the future predicted under a candidate action with the environmental future predicted under the common comparison action condition. Here, *paired* means that the two futures are generated from the same current observation and task instruction and are evaluated by the same future evaluator. Within a candidate group, the same environmental future and its evaluation are shared across all candidate actions. The two generations are not required to use the same random seed, latent noise, or stochastic realization.

The causal target in **Eq. 32** is a reference-relative conditional average causal contrast between two observation-conditioned interventional expectations. It is therefore different from a unit-level counterfactual comparison that holds a particular realization of the exogenous variables fixed across interventions. The stochastic variables of the learned future generator are also not assumed to correspond one-to-one to the exogenous environmental factors in the SCM.

We define the model-level expected future evaluation under action condition $\mathbf{a}$ as

$$\bar{Y}_{\phi,t} \left(\mathbf{a} \mid \mathbf{o}, \mathbf{c} \right) = \mathbb{E}_{\hat{\mathbf{O}}_{t+1:t+H} \sim p_\phi(\cdot \mid \mathbf{o}, \mathbf{a}, \mathbf{c})} \left[V \left(\hat{\mathbf{O}}_{t+1:t+H}, \mathbf{c} \right) \right]. \tag{36}$$

For the comparison action condition $\mathbf{a}^{\mathrm{ref}}$, the corresponding model-level expected contrast is

$$\begin{aligned} \bar{G}_{\phi,t} \left(\mathbf{a}, \mathbf{a}^{\mathrm{ref}} \mid \mathbf{o}, \mathbf{c} \right) = \bar{Y}_{\phi,t} \left(\mathbf{a} \mid \mathbf{o}, \mathbf{c} \right) \\ - \bar{Y}_{\phi,t} \left(\mathbf{a}^{\mathrm{ref}} \mid \mathbf{o}, \mathbf{c} \right). \end{aligned} \tag{37}$$

In practice, CAG-WAM evaluates sampled futures from the two action conditions. For candidate action $\mathbf{A}_t^i$, the realized causal action gain is

$$G_t^i = Y_t^i - Y_t^{\mathrm{env}}. \tag{38}$$

Taking expectation over the stochastic future generation gives

$$\mathbb{E}_\phi \left[G_t^i \mid \mathbf{O}_t = \mathbf{o}, \mathbf{c}, \mathbf{A}_t^i = \mathbf{a} \right] = \bar{G}_{\phi,t} \left(\mathbf{a}, \mathbf{a}^{\mathrm{ref}} \mid \mathbf{o}, \mathbf{c} \right). \tag{39}$$

This equality follows from linearity of expectation and does not require shared generation noise. Each realized G_t^i is therefore a stochastic sample-level comparison between the candidate and comparison action conditions, whose expectation is the model-level contrast in **Eq. 37**.

Under the identification, support, and model-approximation assumptions in **Appendix C.1**, the model-level contrast in **Eq. 37** provides a model-based approximation to the reference-relative conditional average causal contrast in **Eq. 32**.

Approximation error of paired-future evaluation. The practical causal action gain depends jointly on the learned future predictor and future evaluator. Rather than assuming that their errors can be separated, we characterize their combined effect through the action-conditioned evaluation approximation error. Define

$$\varepsilon_{\phi,t} \left(\mathbf{a} \mid \mathbf{o}, \mathbf{c} \right) = \bar{Y}_{\phi,t} \left(\mathbf{a} \mid \mathbf{o}, \mathbf{c} \right) - \mathbb{E} \left[Y_t \mid do(\mathbf{A}_t = \mathbf{a}), \mathbf{O}_t = \mathbf{o}, \mathbf{c} \right]. \tag{40}$$

Using Eqs. 37 and 32, the difference between the model-level expected contrast and the target conditional average causal contrast is

$$\begin{aligned} &\bar{G}_{\phi,t} \left(\mathbf{a}, \mathbf{a}^{\mathrm{ref}} \mid \mathbf{o}, \mathbf{c} \right) - \tau_t \left(\mathbf{a}, \mathbf{a}^{\mathrm{ref}} \mid \mathbf{o} \right) \\ &\qquad = \varepsilon_{\phi,t} \left(\mathbf{a} \mid \mathbf{o}, \mathbf{c} \right) - \varepsilon_{\phi,t} \left(\mathbf{a}^{\mathrm{ref}} \mid \mathbf{o}, \mathbf{c} \right). \end{aligned} \tag{41}$$

Equation 41 shows that approximation errors shared across the two action conditions may partially cancel, but such cancellation is not guaranteed. In particular, differential prediction or evaluation errors between the candidate and comparison action conditions can bias the model-level causal action gain. This is why the support and model-approximation assumptions in **Appendix C.1** apply to both sides of the paired-future comparison.

Cancellation of shared evaluation components. Suppose an additive evaluation component C_t contributes equally to the action-conditioned and environmental futures under the matched observation and task conditions, such that

$$Y_t^i = C_t + \widetilde{Y}_t^i, \qquad Y_t^{\mathrm{env}} = C_t + \widetilde{Y}_t^{\mathrm{env}}, \tag{42}$$

where $\widetilde{Y}_t^i$ and $\widetilde{Y}_t^{\mathrm{env}}$ denote the remaining parts of the two evaluations. Substituting **Eq. 42** into **Eq. 38** gives

$$G_t^i = \widetilde{Y}_t^i - \widetilde{Y}_t^{\mathrm{env}}. \tag{43}$$

Thus, any additive component that contributes equally to the two realized evaluations is canceled exactly in their difference. This property does not require the full future evaluation to admit an additive decomposition; it applies only to components that are shared by the two evaluations.

Relation to candidate-relative comparison. Because the same environmental future evaluation Y_t^{env} is shared across the candidate group, for two candidate actions $\mathbf{A}_t^i$ and $\mathbf{A}_t^j$ we have

$$\begin{aligned} G_t^i - G_t^j &= \left(Y_t^i - Y_t^{\mathrm{env}} \right) - \left(Y_t^j - Y_t^{\mathrm{env}} \right) \\ &= Y_t^i - Y_t^j. \end{aligned} \tag{44}$$

Therefore, subtracting the shared environmental future does not change the relative ordering among candidate actions. This invariance is expected: the role of the environmental future is not to produce a different within-group ranking. Rather, retaining it allows each candidate evaluation to remain expressed relative to the same environmental future. Candidate-relative comparison and causal action gain therefore contain the same pairwise ordering information, while differing in whether this common comparison term is retained.

Effect of group centering. This distinction becomes explicit under group centering. Let

$$\bar{Y}_t = \frac{1}{K}\sum_{i=1}^{K} Y_t^i, \qquad \bar{G}_t = \frac{1}{K}\sum_{i=1}^{K} G_t^i. \tag{45}$$

Using **Eq. 38**, we have

$$\bar{G}_t = \bar{Y}_t - Y_t^{\mathrm{env}}. \tag{46}$$

Consequently,

$$\begin{aligned} G_t^i - \bar{G}_t &= \left(Y_t^i - Y_t^{\mathrm{env}}\right) - \left(\bar{Y}_t - Y_t^{\mathrm{env}}\right) \\ &= Y_t^i - \bar{Y}_t. \end{aligned} \tag{47}$$

Equation 47 shows that group centering cancels the environmental future exactly and reduces the signal to a candidate-relative comparison. CAG-WAM therefore retains the uncentered causal action gain, so that the sign and magnitude of each candidate evaluation remain defined relative to the environmental future rather than only relative to the other sampled candidates.

Thus, paired future evaluation preserves the common environmental comparison without changing the pairwise ordering among candidates.

C.3 CAG-Guided Policy Optimization

Under the identification, support, and model-approximation assumptions in **Appendix C.1**, the model-level expectation of causal action gain approximates the conditional average causal contrast defined there. CAG-WAM uses the realized causal action gain of each sampled candidate directly as its optimization advantage,

$$\Delta_{t,\mathrm{CAG}}^i = G_t^i = Y_t^i - Y_t^{\mathrm{env}}. \tag{48}$$

During each policy update, the paired future predictions and their evaluations are treated as fixed rollout feedback, and gradients are taken only with respect to the action policy. The causal action gain therefore weights the sampled action trajectory, while its causal interpretation follows from the action-evaluation analysis in **Appendix C.1**.

Expected-gradient interpretation. Conditioned on the current observation $\mathbf{O}_t$, task instruction $\mathbf{c}$, and a realized shared environmental evaluation Y_t^{env}, consider

$$\mathcal{J}_Y(\theta) = \mathbb{E}_{\mathbf{A}_t \sim \pi_\theta(\cdot|\mathbf{O}_t,\mathbf{c})}\left[Y_t(\mathbf{A}_t)\right]. \tag{49}$$

Using the score-function identity reviewed in **Appendix B.4**,

$$\nabla_\theta \mathcal{J}_Y(\theta) = \mathbb{E}\left[Y_t(\mathbf{A}_t)\nabla_\theta \log \pi_\theta(\mathbf{A}_t \mid \mathbf{O}_t, \mathbf{c})\right]. \tag{50}$$

For the same current observation and task instruction, Y_t^{env} is shared across candidate actions and does not depend on which candidate action is sampled. It therefore acts as an action-independent baseline,

$$\mathbb{E}\left[Y_t^{\mathrm{env}}\nabla_\theta \log \pi_\theta(\mathbf{A}_t \mid \mathbf{O}_t, \mathbf{c})\right] = 0. \tag{51}$$

Combining Eqs. 50 and 51 gives

$$\nabla_\theta \mathcal{J}_Y(\theta) = \mathbb{E}\left[G_t(\mathbf{A}_t)\nabla_\theta \log \pi_\theta(\mathbf{A}_t \mid \mathbf{O}_t, \mathbf{c})\right]. \tag{52}$$

Thus, subtracting the shared environmental evaluation does not change the ideal expected score-function gradient. The causal interpretation of CAG comes from the action-evaluation analysis in **Appendix C.1**, while the resulting gain is used here as the policy-optimization signal.

Flow-based finite-sample optimization. As reviewed in **Appendix B.3**, Flow-GRPO represents the flow-matching policy as a sequence of stochastic transitions with tractable transition likelihoods. Using the trajectory-gradient decomposition in **Eq. 18**, the action-level advantage $\Delta_{t,\mathrm{CAG}}^i$ is applied to the stochastic transitions of the generation trajectory for candidate action $\mathbf{A}_t^i$. For K sampled candidates, the corresponding finite-sample transition-level score-function term is

$$\widehat{\mathbf{g}}_{\mathrm{CAG}} = \frac{1}{K}\sum_{i=1}^{K}\frac{1}{T}\sum_{\tau=1}^{T}\Delta_{t,\mathrm{CAG}}^i \nabla_\theta \log p_\theta\left(\mathbf{X}_{t,\tau-1}^i \mid \mathbf{X}_{t,\tau}^i, \mathbf{O}_t, \mathbf{c}\right). \tag{53}$$

The baseline identity in **Eq. 52** holds after expectation over policy sampling. For a finite sampled group, the empirical average of the score-function terms need not be exactly zero. The shared environmental evaluation can therefore affect the realized update even though its expected contribution vanishes. As shown in **Appendix C.2**, additional group-mean centering would instead cancel the environmental evaluation exactly and reduce the advantage to a candidate-relative comparison. CAG-WAM retains the uncentered causal action gain so that the sign and magnitude of each realized gain remain defined relative to the environmental future.

Clipped policy optimization. Following the GRPO objective and its flow-based formulation reviewed in Appendices B.2 and B.3, we optimize the action policy with

$$\mathcal{J}_{\mathrm{CAG}}(\theta) = \mathbb{E}_{\theta_{\mathrm{old}}}\Bigg[\frac{1}{K}\sum_{i=1}^{K}\frac{1}{T}\sum_{\tau=1}^{T}\Big(\min\big(\rho^i_{t,\tau}(\theta)\Delta^i_{t,\mathrm{CAG}}, \mathrm{clip}(\rho^i_{t,\tau}(\theta), 1-\epsilon, 1+\epsilon)\Delta^i_{t,\mathrm{CAG}}\big) - \beta D_{\mathrm{KL}}(\pi_\theta \| \pi_{\mathrm{ref}})\Big)\Bigg], \tag{54}$$

where $\rho^i_{t,\tau}(\theta)$ is the transition-level importance ratio defined in **Eq. 19**, ϵ is the clipping coefficient, β controls the KL regularization, and π_{ref} denotes the reference policy.

At the sampling policy, $\theta = \theta_{\mathrm{old}}$, we have $\rho^i_{t,\tau}(\theta) = 1$. Since 1 lies inside the clipping interval, the clipped and unclipped terms coincide locally,

$$\begin{aligned}\nabla_\theta \min\big(\rho^i_{t,\tau}(\theta)\Delta^i_{t,\mathrm{CAG}}, \mathrm{clip}(\rho^i_{t,\tau}(\theta), 1-\epsilon, 1+\epsilon)\Delta^i_{t,\mathrm{CAG}}\big)\big|_{\theta=\theta_{\mathrm{old}}} \\ = \Delta^i_{t,\mathrm{CAG}}\ \nabla_\theta \log p_\theta\left(\mathbf{X}^i_{t,\tau-1} \mid \mathbf{X}^i_{t,\tau}, \mathbf{O}_t, \mathbf{c}\right)\big|_{\theta=\theta_{\mathrm{old}}}.\end{aligned} \tag{55}$$

Around the sampling policy, positive realized causal action gains locally favor their sampled generation transitions, while negative gains produce the opposite update.

Away from the sampling policy, the clipped surrogate is not invariant to an action-independent shift of the advantage. The shared environmental term can therefore affect the magnitude of individual surrogate terms and whether they are limited by clipping. Together with finite-sample estimation, this allows the realized CAG update to differ from an update based only on candidate-future evaluations, despite the same ideal expected score-function gradient.

Fixed-model optimization and policy shift. The PF-WM and future evaluator remain fixed while the action policy is updated. Policy updates may therefore move candidate actions into regions where future prediction or evaluation is less accurate, in which case the resulting causal action gain inherits these approximation errors. Condition-dependent errors are particularly relevant when the errors for the candidate and comparison action conditions differ, as characterized in **Appendix C.2**.

Low-rank policy adaptation, ratio clipping, and KL regularization toward the reference policy constrain policy shift from the pretrained action distribution, but they do not guarantee the accuracy of the fixed PF-WM and future evaluator. The reliability of the resulting causal action gain therefore remains subject to the support and model-approximation assumptions in **Appendix C.1**.

C.4 Interpretation of Future Feedback and Action Revision

For an initial action chunk $\mathbf{A}^{\mathrm{init}}_t$, CAG-WAM predicts the corresponding action-conditioned future representation $\hat{\mathbf{Z}}^{\mathrm{init}}_{t+1:t+H}$ and environmental future representation $\hat{\mathbf{Z}}^{\mathrm{env}}_{t+1:t+H}$. Their evaluations give the initial causal action gain

$$G^{\mathrm{init}}_t = Y^{\mathrm{init}}_t - Y^{\mathrm{env}}_t. \tag{56}$$

This gain determines whether an additional action generation is invoked. Future Revision is activated when

$$G^{\mathrm{init}}_t < \delta, \tag{57}$$

where δ controls the selectivity of the revision criterion.

A negative G_t^{init} means that the initial action-conditioned future receives a lower evaluation than the environmental future. A positive δ introduces a practical revision threshold, so revision may also be activated when the action-conditioned future receives a slightly higher evaluation but its gain remains below the selected threshold. The value of δ is defined on the scale of the future evaluator and is treated as an implementation hyperparameter rather than a universal causal boundary. When $\delta = 0$, the decision depends only on the sign of the causal action gain. We use $\delta = 0.05$ in our experiments and analyze its sensitivity in **Appendix I.5**.

The scalar gain and the representation-level feedback are obtained from the same paired future predictions but serve different roles. G_t^{init} determines whether revision is invoked. Once revision is activated, the future representations provide the additional conditioning information for a second action generation. The feedback rule described below is a design choice for the FC-AM conditioning interface rather than a causal intervention derived from the SCM.

Future-feedback construction. For an initial action satisfying **Eq. 57**, we define the representation-level discrepancy between the action-conditioned and environmental futures as

$$\mathbf{D}_t = \hat{\mathbf{Z}}^{\text{init}}_{t+1:t+H} - \hat{\mathbf{Z}}^{\text{env}}_{t+1:t+H}. \tag{58}$$

The future feedback is constructed as

$$\hat{\mathbf{Z}}^{\text{fb}}_{t+1:t+H} = \hat{\mathbf{Z}}^{\text{init}}_{t+1:t+H} + \gamma \mathbf{D}_t, \qquad \gamma \geq 0, \tag{59}$$

where γ controls the contribution of the paired-future discrepancy.

This residual form preserves the dimensionality and conditioning interface of the initial future representation while incorporating information that distinguishes the action-conditioned future from the environmental future. When $\gamma = 0$, the feedback reduces to the initial action-conditioned future representation, while larger values place greater weight on their representation-level discrepancy.

The discrepancy $\mathbf{D}_t$ describes how the two predicted futures differ in representation space. We do not assume that its direction corresponds to an increase or decrease in future evaluation, or to a desired change in the robot action. Accordingly, $\hat{\mathbf{Z}}^{\text{fb}}_{t+1:t+H}$ is used only as additional conditioning information for a new action generation. It is not treated as a target future representation or directly applied as an action update. The FC-AM performs a complete new generation conditioned on the current observation, task instruction, and future feedback.

Action regeneration and fusion. Let $\mathbf{A}_t^{\text{rev}}$ denote the action chunk produced by the additional future-conditioned generation. The revised action is generated by the FC-AM rather than obtained by directly applying $\mathbf{D}_t$ to the initial action. The final executed action is

$$\mathbf{A}_t^{\text{exec}} = \mathbf{A}_t^{\text{init}} + \lambda \left(\mathbf{A}_t^{\text{rev}} - \mathbf{A}_t^{\text{init}} \right), \qquad \lambda \in [0, 1]. \tag{60}$$

Equivalently,

$$\mathbf{A}_t^{\text{exec}} = (1 - \lambda)\mathbf{A}_t^{\text{init}} + \lambda \mathbf{A}_t^{\text{rev}}. \tag{61}$$

The coefficient λ controls the contribution of the revised action to execution. When $\lambda = 0$, the initial action is retained, while $\lambda = 1$ uses the revised action entirely. The initial and revised action chunks share the same action representation, horizon, and temporal indexing, so the fusion is applied element-wise between corresponding action steps.

D PSEUDO-CODE

In this section, we provide the pseudo-code of CAG-WAM. The overall procedure is shown in **Algorithm 1** and **Algorithm 2**.

E BENCHMARKS

In this section, we briefly describe the robotic manipulation benchmarks used in our experiments.

Algorithm 1 Causal Action Gain-Guided Policy Optimization

Require: Current observation $\mathbf{O}_t$, task instruction $\mathbf{c}$, sampling policy $\pi_{\theta_{\text{old}}}$, action policy π_θ, Paired-Future World Model $\mathcal{W}$, number of candidates K
Ensure: Updated policy parameters θ
1: Generate K candidate action chunks

$$\{\mathbf{A}_t^i\}_{i=1}^K \sim \pi_{\theta_{\text{old}}}(\cdot \mid \mathbf{O}_t, \mathbf{c})$$

2: Predict the environmental future and its evaluation

$$(\hat{\mathbf{O}}^{\text{env}}_{t+1:t+H}, Y_t^{\text{env}}) \leftarrow \mathcal{W}(\mathbf{O}_t, \mathbf{c}, \text{HOLD})$$

3: **for** $i = 1, \ldots, K$ **do**
4: Predict the action-conditioned future and its evaluation

$$(\hat{\mathbf{O}}^{i}_{t+1:t+H}, Y_t^{i}) \leftarrow \mathcal{W}(\mathbf{O}_t, \mathbf{c}, \mathbf{A}_t^i)$$

5: Compute causal action gain

$$G_t^i \leftarrow Y_t^i - Y_t^{\text{env}}$$

6: Set the policy advantage

$$\Delta^i_{t,\text{CAG}} \leftarrow G_t^i$$

7: **end for**
8: Update π_θ using the clipped CAG-guided GRPO objective with $\{\Delta^i_{t,\text{CAG}}\}_{i=1}^K$
9: **return** θ

- RoboCasa (Nasiriany et al., 2024) is a simulated robotic manipulation benchmark built around diverse kitchen environments. It contains tasks involving object manipulation, articulated-object interaction, and other common household behaviors, with variations in objects, scene layouts, and interaction conditions. We evaluate on the 24-task manipulation setting and group the tasks into seven skill families for result reporting: Pick&Place, Door, Drawer, Knob, Lever, Button, and Insertion. We use RoboCasa to evaluate CAG-WAM across diverse manipulation skills and kitchen interactions.
- LIBERO (Liu et al., 2023) is a robotic manipulation benchmark containing four standard task suites: LIBERO-Spatial, LIBERO-Object, LIBERO-Goal, and LIBERO-Long. Each suite contains 10 tasks and emphasizes different manipulation requirements, including spatial relations, object variations, task goals, and long-horizon execution, respectively. We evaluate on all four suites to further examine the effectiveness of CAG-WAM across different task settings.

F BASELINES

In this section, we provide a brief overview of the baselines used for comparison in our experiments.

- 3D Diffuser Actor (Ke et al., 2024) is a diffusion-based robotic manipulation policy that performs action generation using 3D scene representations. It incorporates geometric information from the observed environment into the diffusion process to model continuous robot actions.
- DP3 (Ze et al., 2024) is a 3D diffusion policy for robotic manipulation. It represents the environment using compact point-cloud features and combines them with diffusion-based action generation for continuous robot control.
- Diffusion Policy (Chi et al., 2025) formulates visuomotor policy learning as conditional diffusion over robot action sequences. Given the current observations, the policy iteratively denoises an action trajectory and produces continuous actions for execution. We consider both the ResNet- and CLIP-based visual encoder variants in the RoboCasa experiments.
- GR00T N1 (Bjorck et al., 2025) is a generalist robot foundation model designed for diverse embodied reasoning and manipulation tasks. It combines visual and language information with continuous action generation and supports multiple robot embodiments.

Algorithm 2 Causal Action Gain-Guided Future Revision

Require: Current observation $\mathbf{O}_t$, task instruction $\mathbf{c}$, action policy π_θ, Paired-Future World Model $\mathcal{W}$, revision threshold δ, feedback coefficient γ, fusion coefficient λ

Ensure: Executed action chunk $\mathbf{A}_t^{\text{exec}}$

1: Generate the initial action chunk

$$\mathbf{A}_t^{\text{init}} \leftarrow \pi_\theta(\mathbf{O}_t, \mathbf{c})$$

2: Predict the action-conditioned future representation and evaluation

$$(\hat{\mathbf{Z}}_{t+1:t+H}^{\text{init}}, Y_t^{\text{init}}) \leftarrow \mathcal{W}(\mathbf{O}_t, \mathbf{c}, \mathbf{A}_t^{\text{init}})$$

3: Predict the environmental future representation and evaluation

$$(\hat{\mathbf{Z}}_{t+1:t+H}^{\text{env}}, Y_t^{\text{env}}) \leftarrow \mathcal{W}(\mathbf{O}_t, \mathbf{c}, \text{HOLD})$$

4: Compute the initial causal action gain

$$G_t^{\text{init}} \leftarrow Y_t^{\text{init}} - Y_t^{\text{env}}$$

5: **if** $G_t^{\text{init}} \geq \delta$ **then**

6:

$$\mathbf{A}_t^{\text{exec}} \leftarrow \mathbf{A}_t^{\text{init}}$$

7: **else**

8: Compute the paired-future representation discrepancy

$$\mathbf{D}_t \leftarrow \hat{\mathbf{Z}}_{t+1:t+H}^{\text{init}} - \hat{\mathbf{Z}}_{t+1:t+H}^{\text{env}}$$

9: Construct future feedback

$$\hat{\mathbf{Z}}_{t+1:t+H}^{\text{fb}} \leftarrow \hat{\mathbf{Z}}_{t+1:t+H}^{\text{init}} + \gamma \mathbf{D}_t$$

10: Perform future-conditioned action generation

$$\mathbf{A}_t^{\text{rev}} \leftarrow \pi_\theta\left(\mathbf{O}_t, \mathbf{c}, \hat{\mathbf{Z}}_{t+1:t+H}^{\text{fb}}\right)$$

11: Fuse the initial and revised actions

$$\mathbf{A}_t^{\text{exec}} \leftarrow \mathbf{A}_t^{\text{init}} + \lambda\left(\mathbf{A}_t^{\text{rev}} - \mathbf{A}_t^{\text{init}}\right)$$

12: **end if**

13: **return** $\mathbf{A}_t^{\text{exec}}$

- FPV (Donat et al., 2025) is an imitation-learning method that combines RGB observations with point-cloud representations for robotic manipulation. Its SUGAR representation integrates complementary visual and geometric information for policy learning.
- Octo (Team et al., 2024) is an open-source generalist robot policy trained on large-scale and heterogeneous robot datasets. It uses a unified policy architecture to support different manipulation tasks, observations, and robot embodiments.
- OpenVLA (Kim et al., 2024) is an open-source Vision-Language-Action model that adapts pretrained vision-language representations to robot control. It takes visual observations and language instructions as input and predicts continuous robot actions.
- TraceVLA (Zheng et al., 2025) extends Vision-Language-Action modeling with visual trace prompting. The visual traces provide additional spatial-temporal information about manipulation trajectories and object interactions.
- SpatialVLA (Qu et al., 2025) incorporates explicit spatial information into Vision-Language-Action learning. It strengthens the representation of object locations, spatial relations, and manipulation geometry for robot action generation.
- UniVLA (Bu et al., 2025) introduces task-centric latent actions for Vision-Language-Action learning. It learns latent action representations that capture task-relevant manipulation behaviors and uses them to support action generation across different tasks.

- UVA (Li et al., 2025) is a unified video-action model that jointly models visual dynamics and robot actions. It integrates video prediction and action learning within a shared framework so that temporal visual information can be used during robot policy learning.
- Video Policy (Liang et al., 2025) adapts pretrained video-generation models to robotic control. It leverages the spatiotemporal representations learned by video models for continuous action prediction. We consider the variants trained with 50 and 300 demonstrations per task in the RoboCasa experiments.
- LingBot-VA (Li et al., 2026) is a video-action model that jointly models visual dynamics and robot actions for closed-loop manipulation. It combines video-based predictive modeling with continuous action generation and updates its predictions using new observations during interaction.
- Cosmos Policy-Predict2 (Kim et al., 2026) adapts the pretrained Cosmos-Predict2 video foundation model to visuomotor control. It jointly models robot actions, future visual states, and future values within the video-generation framework, and can use these predictions for action generation and model-based planning.
- Cosmos Policy-Predict2.5 (Kim et al., 2026) follows the Cosmos Policy formulation using the updated Cosmos-Predict2.5 video foundation model. It retains the joint modeling of robot actions and predicted future information while using the newer video-model backbone.

G Discussion

G.1 Comparison and Uniqueness Discussion

In this section, we further discuss the differences between CAG-WAM and related studies from the perspectives of future evaluation, causal modeling, policy optimization, and inference-time action generation.

Recent World Action Models explore different ways of connecting future prediction with robot action generation. MotuBrain (Team et al., 2026) jointly models visual dynamics and robot actions within a unified generative framework. τ_0-WM (Zhou et al., 2026) predicts action-conditioned futures and task-progress information to support candidate evaluation and action refinement. Self-WAM (Pan et al., 2026) strengthens the dependence of predicted futures on robot actions through explicit action conditioning and self-supervision, while HiTac-WAM (Xue et al., 2026) further incorporates tactile future prediction for contact-rich manipulation. Cosmos Policy (Kim et al., 2026) jointly models robot actions, future states, and future values with a pretrained video model and uses these predictions for action generation and planning. These studies use predicted futures to support robot control in different ways. CAG-WAM focuses instead on how the future associated with a candidate action is evaluated when the predicted evolution may contain changes related to both the candidate action and environmental evolution. For each candidate action, CAG-WAM predicts an action-conditioned future together with an environmental future under the same current observation, and uses their evaluation difference to characterize the future change associated with that candidate action.

CAG-WAM is also related to causal approaches in robot learning. CREST (Lee et al., 2021) uses interventions in internal models to identify task-relevant variables for manipulation policies. RoCoDA (Ameperosa et al., 2025) uses causal invariance to construct counterfactual demonstrations, while CofactVLA (Zhang et al., 2026) studies visual confounding in VLA policies and introduces counterfactual intervention during action generation. FACT (Peng et al., 2026) introduces causal training into World Action Models by conditioning future prediction on executed actions and incorporating failure trajectories to model action-dependent consequences. CAG-WAM considers causal analysis at the stage of future evaluation in WAMs. We construct an SCM to describe how the current observation, candidate action, action-induced changes, environment-driven changes, and future evaluation are related, and identify the corresponding backdoor path. Based on this analysis, the action-conditioned future and environmental future are predicted under matched current conditions, and their evaluation difference is used to characterize the future change associated with the candidate action. This connects the causal analysis directly to how predicted futures are evaluated and used by the action policy.

Another difference lies in how multiple candidate actions are used during policy optimization. Candidate sampling and group-relative comparison can distinguish relatively better and worse actions within the sampled set, and GRPO provides an efficient framework for policy optimization from grouped samples (Shao et al., 2024; Liu et al., 2026a). CAG-WAM retains the grouped candidate generation setting but changes the information used as the advantage. Instead of evaluating a candidate only relative to the other sampled actions, CAG-WAM compares its predicted future with the corresponding environmental future and uses the resulting causal action gain directly as the advantage. In this way, the policy update can reflect whether the candidate-conditioned future receives a higher or lower evaluation than the environmental future, while retaining the clipped GRPO optimization framework.

Predicted futures have also been used during inference for candidate ranking, planning, and action refinement. For example, τ_0-WM (Zhou et al., 2026) uses predicted future outcomes to support candidate evaluation and rectification, while Cosmos Policy (Kim et al., 2026) uses future-state and value prediction for model-based action search. CAG-WAM also uses predicted futures before execution, but the inference procedure is based on the same paired future evaluation used during training. The initial action is first evaluated through its causal action gain. When the gain falls below the revision threshold δ, the paired future representations are used to construct future feedback for an additional generation by the same action model. The revised action is then combined with the initial action before execution. Thus, the paired future information is used consistently in both policy optimization and action revision.

G.2 Further Illustration of Figure 1

Figure 1a illustrates two cases in which the overall future evaluation may not fully reflect the future change associated with a candidate action. **Figure 1b** summarizes the causal structure used to analyze this issue.

In the first scenario, the environmental future is relatively favorable, with the robot remaining near the target object. Consider two candidate actions that move the robot away from the object, with one causing a larger deviation than the other. Although both actions make the resulting future worse relative to the environmental future, their predicted futures may still receive favorable evaluations overall. Thus, the overall future evaluation alone does not indicate whether a candidate action improves or worsens the future relative to the environmental evolution.

In the second scenario, the environmental future is unfavorable, with the robot remaining far from the target region. Two candidate actions move the robot closer to the target, although neither is sufficient to make the resulting future fully favorable. Their predicted futures may therefore still receive low evaluations, even though both actions improve the outcome relative to the environmental future. Together, these two cases illustrate that the overall future evaluation and the future change associated with the candidate action are related but not necessarily equivalent.

Candidate-relative comparison can distinguish better and worse actions within the sampled set, but it does not by itself indicate whether each candidate improves or worsens the future relative to the environmental evolution. For example, in the first scenario, one harmful candidate may still be preferred over another that moves the robot farther away. Likewise, in the second scenario, one beneficial candidate may receive a lower relative preference than another that moves the robot closer to the target. Our interest is therefore not only in the relative ordering among sampled candidates, but also in how each candidate changes the predicted future relative to the environmental evolution.

Figure 1b formalizes this setting with the SCM. The current observation $\mathbf{O}_t$ affects both the generation of the candidate action $\mathbf{A}_t$ and the subsequent environment-driven changes $\mathbf{E}_{t+1}$. Exogenous environmental factors $\mathbf{U}_t$ can also affect $\mathbf{E}_{t+1}$, while the candidate action gives rise to the action-induced changes $\mathbf{M}_{t+1}$. The environment-driven and action-induced changes jointly determine the future observation $\mathbf{O}_{t+1}$, which is subsequently evaluated as Y_t. Because $\mathbf{O}_t$ influences both the candidate action and the environment-driven changes, the candidate action and future evaluation are also connected through the backdoor path $\mathbf{A}_t \leftarrow \mathbf{O}_t \rightarrow \mathbf{E}_{t+1} \rightarrow \mathbf{O}_{t+1} \rightarrow Y_t$. The variables $\mathbf{M}_{t+1}$ and $\mathbf{E}_{t+1}$ describe different causal sources of future change, but the SCM does not require the predicted observation to be explicitly decomposed into separate components.

This analysis motivates the paired-future comparison in CAG-WAM. For the same current observation and task instruction, we predict an action-conditioned future under the candidate action and an environmental future describing how the scene may evolve without applying that candidate-specific action. Their evaluations are computed by the same future evaluator, and their difference defines the causal action gain. During training, this gain is used as the policy-optimization advantage. During inference, it determines whether Future Revision is invoked, while the paired future representations provide the conditioning information for the additional action generation.

G.3 Temporal Abstraction of the SCM

The SCM in **Section 3** is defined at the decision level rather than at the level of individual low-level control steps. Specifically, the action variable $\mathbf{A}_t$ denotes the entire candidate action chunk considered by the WAM at decision time t,

$$\mathbf{A}_t = [\mathbf{a}_t, \dots, \mathbf{a}_{t+L-1}], \tag{62}$$

and the intervention $do(\mathbf{A}_t = \mathbf{a})$ should therefore be understood as an intervention on this decision-level action-chunk variable. This abstraction is used for analyzing candidate-action evaluation and does not imply that the entire chunk must be executed open-loop before the policy is queried again.

Similarly, the future-observation variable $\mathbf{O}_{t+1}$ in the SCM is a compact decision-level representation of the future outcome associated with the candidate action over the prediction horizon. In the implemented world model, this future outcome is represented by the predicted observation sequence $\hat{\mathbf{O}}_{t+1:t+H}$, which is subsequently evaluated by the future evaluator. Accordingly, the directed path $\mathbf{A}_t \to \mathbf{M}_{t+1} \to \mathbf{O}_{t+1}$ in the SCM summarizes how the candidate action chunk can influence the future outcome through action-induced changes over the evaluation horizon, rather than describing only a single low-level state transition.

The environment-driven changes $\mathbf{E}_{t+1}$ and exogenous environmental factors $\mathbf{U}_t$ should be interpreted at the same decision-level temporal abstraction. They summarize environmental evolution and external factors relevant to the future outcome over the corresponding horizon, rather than variables associated with only one physical time step. The action-chunk length L and future-prediction horizon H therefore need not be identical; they describe the action condition and the evaluated future horizon, respectively.

G.4 Interpretation of Action-Induced and Environment-Driven Changes

The SCM uses $\mathbf{M}_{t+1}$ and $\mathbf{E}_{t+1}$ to represent two causal sources of future change. $\mathbf{M}_{t+1}$ denotes changes that are causally downstream of the current candidate action, whereas $\mathbf{E}_{t+1}$ denotes changes that are not caused by that action and may instead depend on the current observation and exogenous environmental factors. These variables are causal abstractions and need not correspond to separately observable components of the predicted future.

Action-induced changes are not limited to robot motion. They can also include downstream effects caused by the candidate action, such as object motion following contact or other changes in the scene that occur because the action was applied. Environment-driven changes instead describe evolution that is not initiated by the current candidate action, such as passive scene changes or external disturbances.

The two sources need not combine additively or independently. Their effects may interact when determining the future observation, as represented by

$$\mathbf{O}_{t+1} = f_O\left(\mathbf{E}_{t+1}, \mathbf{M}_{t+1}\right), \tag{63}$$

where f_O may be nonlinear and may capture interactions between $\mathbf{E}_{t+1}$ and $\mathbf{M}_{t+1}$. The SCM therefore does not require a unique physical decomposition of the future observation into action-induced and environment-driven components.

CAG-WAM does not explicitly observe or recover $\mathbf{M}_{t+1}$ and $\mathbf{E}_{t+1}$. Instead, the causal analysis operates through interventions on the action variable and compares the corresponding action-conditioned outcomes, as described in **Appendix C.1**. The environmental future used by CAG-WAM should likewise not be identified with $\mathbf{E}_{t+1}$ alone. It is instead the future predicted under the specified comparison action condition, as discussed in **Appendix G.5**.

G.5 Interpretation of the Environmental Future

The environmental future provides a common comparison for candidate actions under the same current observation and task instruction. In the reference-relative conditional average causal contrast in **Eq. 32**, it corresponds to the future under the specified comparison action condition $\mathbf{a}^{\mathrm{ref}}$. The causal action gain therefore compares the predicted future evaluation under a candidate action with that under the common comparison action condition while keeping the current observation and task instruction fixed.

It is a future predicted under the specified comparison action condition and may contain scene changes associated with that condition together with environmental evolution. Its role is to provide the comparison term in the reference-relative causal contrast rather than to explicitly recover the environment-driven component of the SCM.

The candidate action and the comparison action condition are both assumed to lie within the relevant action-conditioning support of the learned world model. Because the action space is continuous, this requirement refers to the support of the conditional action density rather than positive probability at an individual action value. Predictions outside this support should be regarded as model extrapolation and may provide a less reliable approximation to the causal contrast. Differences in prediction or evaluation error between the candidate and comparison action conditions can likewise affect the resulting gain, as discussed in **Appendix C.2**.

In the SCM, $\mathbf{M}_{t+1}$ and $\mathbf{E}_{t+1}$ describe different causal sources of future change rather than components that CAG-WAM explicitly recovers. The paired-future formulation instead compares the outcome under the candidate action with the outcome under the common comparison action condition. The algebraic properties in **Appendix C.2** and the optimization analysis in **Appendix C.3** require the environmental future to be shared across candidate comparisons, but do not require it to be a pure environment-only trajectory.

In our experiments, we instantiate $\mathbf{a}^{\mathrm{ref}}$ with a stationary HOLD condition. HOLD is a robot control condition rather than the absence of a robot action. Accordingly, the environmental future used in our implementation describes the predicted scene evolution under HOLD when the candidate-specific action is not applied.

H Implementation Details

We implement CAG-WAM by instantiating its Future-Conditioned Action Model (FC-AM) and Paired-Future World Model (PF-WM) on open-source world-action model codebases. For RoboCasa, we use the Cosmos-Predict2.5-2B setting, while the LIBERO implementation follows the corresponding Cosmos-Predict2-2B setting.

The FC-AM is initialized from the pretrained action-generation pathway and adapted with LoRA, while the pretrained backbone remains frozen. It serves as the trainable action policy for both the initial action generation and the additional generation used in Future Revision. We set the LoRA rank to 8 and use bf16 for backbone computation and fp32 for the LoRA parameters and optimizer states. During policy optimization, gradients are applied only to the action policy, while the predicted futures and their evaluations are treated as fixed rollout feedback.

The PF-WM uses the future-prediction and value-estimation capabilities of the pretrained world-action backbone and is not updated by the policy-optimization objective. For each training state, we sample $K = 8$ candidate action chunks from the same current observation and task instruction. An environmental future is generated under a HOLD condition and used as the common reference for the candidate actions, while an action-conditioned future is predicted for each candidate. Here, the term *paired* refers to matching the current observation and task instruction between the action-conditioned and environmental predictions; it does not require the two stochastic generations to use identical random noise. The native value prediction is used as the future evaluator without introducing an additional learned reward model. The resulting causal action gain is directly used as the optimization advantage, without candidate-group mean centering.

The underlying pretrained world-action model represents robot actions, future states, and future-state values within a unified latent formulation. Following the Cosmos Policy training formula-

Table 3: Overview of the 24 RoboCasa manipulation tasks used in our experiments. "PnP" denotes Pick-and-Place, and "Cab" denotes Cabinet. The tasks are grouped into seven manipulation skill families.

Skill Category	Task ID	Task Meaning	Description
Pick&Place	PnPCabToCounter	Cabinet → Counter	Pick an object from an open cabinet and place it on the counter.
	PnPCounterToCab	Counter → Cabinet	Pick an object from the counter and place it inside an open cabinet.
	PnPCounterToMicrowave	Counter → Microwave	Pick an object from the counter and place it inside an open microwave.
	PnPCounterToSink	Counter → Sink	Pick an object from the counter and place it in the sink.
	PnPCounterToStove	Counter → Stove	Pick an object from the counter and place it in a pan or pot on the stove.
	PnPMicrowaveToCounter	Microwave → Counter	Pick an object from an open microwave and place it on the counter.
	PnPSinkToCounter	Sink → Counter	Pick an object from the sink and place it on the counter near the sink.
	PnPStoveToCounter	Stove → Counter	Pick an object from a pan or pot on the stove and place it on the counter.
Door	OpenSingleDoor	Open Single Door	Open a microwave door or a single-door cabinet.
	CloseSingleDoor	Close Single Door	Close a microwave door or a single-door cabinet.
	OpenDoubleDoor	Open Double Door	Open a cabinet with two doors.
	CloseDoubleDoor	Close Double Door	Close a cabinet with two doors.
Drawer	OpenDrawer	Open Drawer	Open a kitchen drawer.
	CloseDrawer	Close Drawer	Close a kitchen drawer.
Knob	TurnOnStove	Turn On Stove	Turn on a specified stove burner by rotating the corresponding stove knob.
	TurnOffStove	Turn Off Stove	Turn off a specified stove burner by rotating the corresponding stove knob.
Lever	TurnOnSinkFaucet	Turn On Sink Faucet	Operate the sink faucet lever to turn on the water flow.
	TurnOffSinkFaucet	Turn Off Sink Faucet	Operate the sink faucet lever to turn off the water flow.
	TurnSinkSpout	Turn Sink Spout	Rotate the sink spout to the target position.
Button	CoffeePressButton	Press Coffee Button	Press the coffee machine button to dispense coffee into the mug.
	TurnOnMicrowave	Turn On Microwave	Turn on the microwave by pressing its start button.
	TurnOffMicrowave	Turn Off Microwave	Turn off the microwave by pressing its stop button.
Insertion	CoffeeSetupMug	Set Up Coffee Mug	Pick the mug from the counter and place it under the coffee machine dispenser.
	CoffeeServeMug	Serve Coffee Mug	Pick the mug from under the coffee machine dispenser and place it on the counter.

tion (Kim et al., 2026), different conditioning patterns are used for policy generation, future prediction, and value estimation:

$$\begin{aligned} \text{Policy:} \quad & p(\mathbf{a}, \mathbf{s}', V(\mathbf{s}') \mid \mathbf{s}), \\ \text{World Model:} \quad & p(\mathbf{s}', V(\mathbf{s}') \mid \mathbf{s}, \mathbf{a}), \\ \text{Value:} \quad & p(V(\mathbf{s}') \mid \mathbf{s}, \mathbf{a}, \mathbf{s}'). \end{aligned} \tag{64}$$

These prediction modes use different conditioning patterns within the same latent generative framework, providing compatible representations for action generation, future prediction, and future evaluation. CAG-WAM builds on this shared latent formulation while introducing its own paired-future evaluation and future-feedback construction.

We optimize the action policy using the clipped GRPO objective with a clipping coefficient of $\epsilon = 0.2$ and a KL regularization coefficient of $\beta = 0.01$. We use AdamW with learning rates of 5×10^{-5} for RoboCasa and 3×10^{-6} for LIBERO. Since the PF-WM remains fixed during policy optimization, we use low-rank LoRA updates, ratio clipping, and KL regularization toward the reference policy to constrain policy drift from the pretrained action distribution while the fixed future

predictor and evaluator provide rollout feedback. Action generation uses five flow denoising steps. For RoboCasa, the policy takes three 224×224 RGB views together with a 9-D proprioceptive state and predicts a 7-D action chunk of length 32. For LIBERO, two RGB views and a 9-D proprioceptive state are used, and the policy predicts a 7-D action chunk of length 16. In both benchmarks, the first 16 low-level actions are executed before the policy is queried again, resulting in receding-horizon closed-loop control.

During inference, the FC-AM first generates the initial action chunk from the current observation and task instruction. The PF-WM then predicts the corresponding action-conditioned and environmental futures and evaluates them using the same future evaluator. Future Revision is selectively activated when the resulting causal action gain falls below the revision threshold $\delta = 0.05$. When revision is required, the internal representations of the action-conditioned and environmental futures are used to construct the future feedback with $\gamma = 1.0$ as described in **Section 4.3**. The future feedback remains in the future-representation space and is provided to the same FC-AM as an additional condition together with the current observation and task instruction. The scalar causal action gain itself is not used as an input to the action model, and no separate correction network is introduced. Instead, the FC-AM performs another complete five-step action generation to obtain the revised action. The specific future-feedback construction is introduced by CAG-WAM as an inference-time conditioning mechanism rather than as a direct action-space correction. The revised action is then combined with the initial action using the action-blending coefficient $\lambda = 0.30$ before execution. The initial and revised action chunks share the same action representation, horizon, and temporal indexing. Action fusion is performed element-wise between the corresponding action steps in the policy action space. The resulting action chunk is then executed using the same action processing and receding-horizon control scheme as the original policy.

For evaluation, we follow the standard protocols of the corresponding benchmarks. On RoboCasa, we evaluate all 24 manipulation tasks with 50 trials per task. On LIBERO, we evaluate the four standard suites, with 50 trials for each task under three different random seeds. Task success rate is used as the primary metric for both benchmarks. All experiments are conducted on NVIDIA H100 80GB GPUs.

I More Experiments and results

I.1 Full Results

We provide additional task-level results on RoboCasa to complement the skill-level results reported in the main paper. RoboCasa contains 24 manipulation tasks covering seven skill categories, including Pick&Place, Door, Drawer, Knob, Lever, Button, and Insertion. Since several task names use abbreviated forms, **Table 3** summarizes the category, task ID, and task meaning for all 24 tasks. For example, "PnP" denotes Pick-and-Place and "Cab" denotes Cabinet. This table provides a more direct interpretation of the individual tasks used in the evaluation.

Table 4 further reports the success rate of each method on every RoboCasa task. The tasks are organized using the same seven skill categories as in **Table 3**, making it easier to examine the results across different manipulation skills. While the main paper reports category-level results for a compact comparison, the task-level results provide a more detailed view of the performance on individual manipulation tasks.

I.2 Controlled Environmental-Dynamics Results.

We construct a controlled environmental-dynamics test on four RoboCasa pick-and-place tasks: PnPCounterToCab, PnPCounterToMicrowave, PnPCounterToSink, and PnPCounterToStove. Before the first robot–object contact, we introduce a short external perturbation to the manipulated object, causing a small passive displacement independently of the robot action. We consider two perturbation directions. In the *Target-Directed* condition, the object is displaced along the horizontal direction toward the task target, representing a favorable environmental change. In the *Target-Opposed* condition, the same displacement is applied in the opposite direction, representing an unfavorable environmental change. The displacement magnitude is fixed to approximately 2–3 cm and is limited such that the object remains in the source workspace and the perturbation alone cannot satisfy the task goal. The perturbation is completed before robot–object contact, and the same protocol is used

Table 4: Task-level success rates (%) on the 24 RoboCasa manipulation tasks. 3DA denotes 3D Diffuser Actor; DP-R and DP-C denote Diffusion Policy with ResNet and CLIP visual encoders, respectively; VP-50 and VP-300 denote Video Policy trained with 50 and 300 demonstrations per task, respectively; Cosmos denotes the Cosmos Policy-Predict2.5.

Task	3DA	DP3	DP-R	DP-C	GR00T	FPV	UVA	VP-50	VP-300	Cosmos	CAG-WAM
Pick&Place											
PnPCabToCounter	0.0	4.0	6.0	0.0	20.0	10.0	26.0	36.0	48.0	44.0	**54.0**
PnPCounterToCab	0.0	2.0	6.0	2.0	36.0	14.0	18.0	42.0	52.0	60.0	**70.0**
PnPCounterToMicrowave	0.0	6.0	6.0	2.0	13.0	10.0	10.0	**52.0**	22.0	36.0	44.0
PnPCounterToSink	0.0	0.0	14.0	8.0	10.0	8.0	16.0	44.0	48.0	58.0	**60.0**
PnPCounterToStove	0.0	0.0	0.0	2.0	24.0	4.0	16.0	58.0	54.0	72.0	**74.0**
PnPMicrowaveToCounter	0.0	0.0	6.0	6.0	16.0	12.0	18.0	44.0	28.0	**48.0**	44.0
PnPSinkToCounter	0.0	0.0	10.0	22.0	33.0	30.0	38.0	64.0	56.0	70.0	**80.0**
PnPStoveToCounter	0.0	0.0	2.0	6.0	29.0	26.0	24.0	64.0	**70.0**	58.0	60.0
Door											
OpenSingleDoor	0.0	24.0	42.0	32.0	59.0	74.0	54.0	68.0	70.0	**94.0**	90.0
CloseSingleDoor	14.0	62.0	78.0	48.0	83.0	84.0	88.0	**100.0**	90.0	92.0	92.0
OpenDoubleDoor	0.0	20.0	70.0	82.0	15.0	92.0	90.0	96.0	94.0	**98.0**	90.0
CloseDoubleDoor	0.0	56.0	78.0	84.0	75.0	78.0	76.0	**98.0**	90.0	74.0	96.0
Drawer											
OpenDrawer	0.0	36.0	64.0	60.0	79.0	72.0	28.0	46.0	54.0	**92.0**	**92.0**
CloseDrawer	0.0	48.0	82.0	96.0	99.0	94.0	72.0	96.0	**100.0**	**100.0**	**100.0**
Knob											
TurnOnStove	10.0	24.0	38.0	28.0	56.0	66.0	50.0	30.0	50.0	64.0	**68.0**
TurnOffStove	2.0	6.0	16.0	8.0	27.0	20.0	14.0	6.0	4.0	34.0	**36.0**
Lever											
TurnOnSinkFaucet	6.0	32.0	66.0	66.0	63.0	70.0	62.0	**84.0**	76.0	78.0	64.0
TurnOffSinkFaucet	28.0	42.0	68.0	70.0	73.0	78.0	64.0	78.0	92.0	88.0	**100.0**
TurnSinkSpout	26.0	54.0	62.0	26.0	53.0	52.0	64.0	40.0	58.0	86.0	**88.0**
Button											
CoffeePressButton	8.0	16.0	76.0	68.0	85.0	90.0	84.0	92.0	**96.0**	**96.0**	94.0
TurnOnMicrowave	6.0	38.0	68.0	88.0	78.0	68.0	94.0	92.0	**96.0**	78.0	82.0
TurnOffMicrowave	32.0	54.0	62.0	**100.0**	71.0	96.0	96.0	90.0	**100.0**	**100.0**	**100.0**
Insertion											
CoffeeSetupMug	0.0	4.0	10.0	12.0	23.0	16.0	20.0	22.0	**58.0**	34.0	28.0
CoffeeServeMug	0.0	18.0	44.0	60.0	73.0	48.0	**78.0**	76.0	70.0	62.0	74.0

Table 5: Success rates (%) under controlled environmental dynamics on four RoboCasa pick-and-place tasks.

Method	Target-Directed	Target-Opposed
Base WAM	54.7	48.9
CAG-WAM	60.5	57.3

for all compared methods. We compare the Base WAM and CAG-WAM using 50 trials per task and report the average success rate over the four tasks. As shown in **Table 5**, CAG-WAM achieves higher average success rates than the Base WAM under both target-directed and target-opposed perturbations. Both methods achieve lower average success rates under the target-opposed condition, suggesting that this condition is more challenging. These results provide additional support for the effectiveness of CAG-WAM.

I.3 More Visualization and Case Study

Figure 5 provides additional qualitative examples of CAG-WAM on three RoboCasa manipulation tasks: "Pick and Place from Counter to Sink", "Pick and Place from Microwave to Counter", and "Pick and Place the Mug Under the Coffee Machine Dispenser". These tasks involve multiple stages

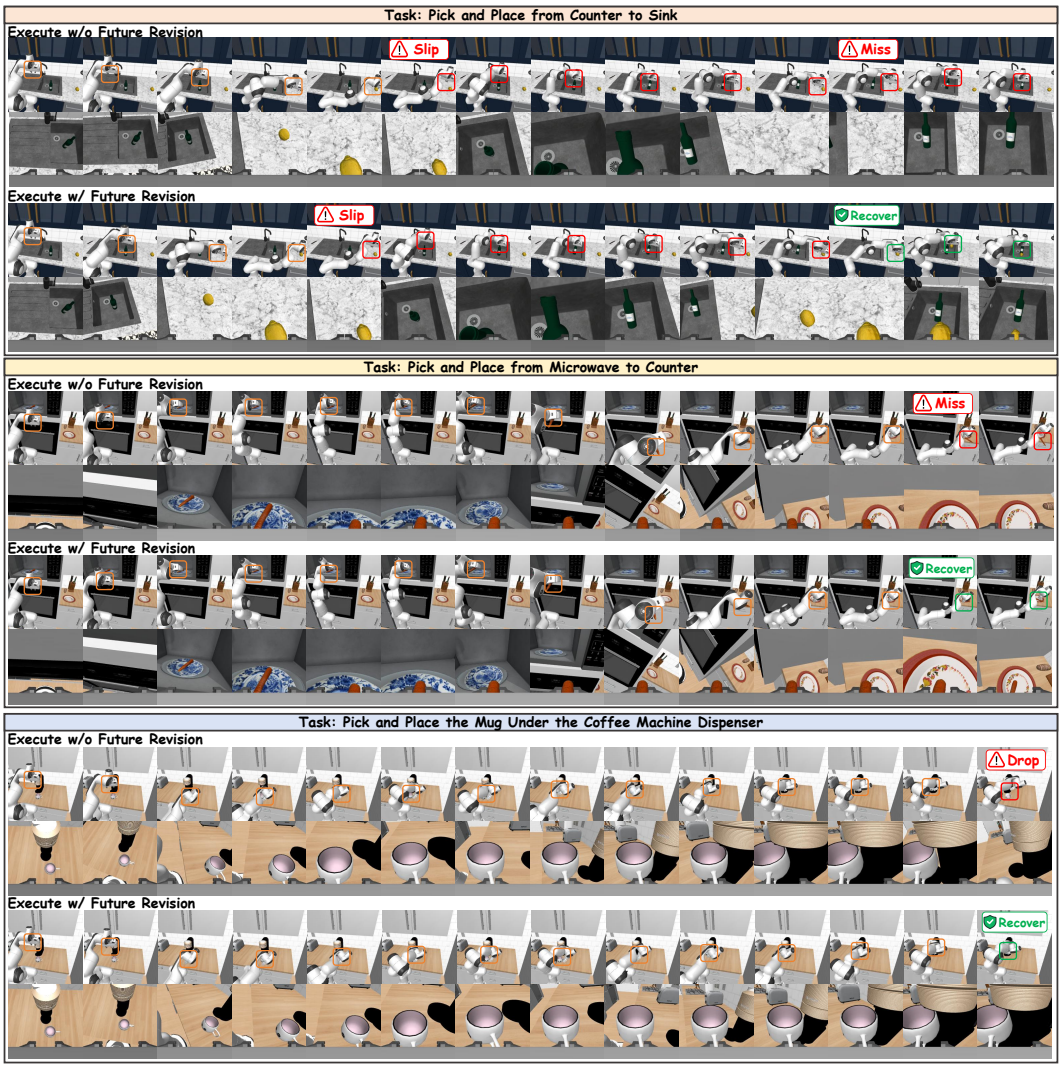


Figure 5: Additional Visualizations of CAG-WAM.

of manipulation, including approaching the object, grasping, transporting, and placing it at the target location.

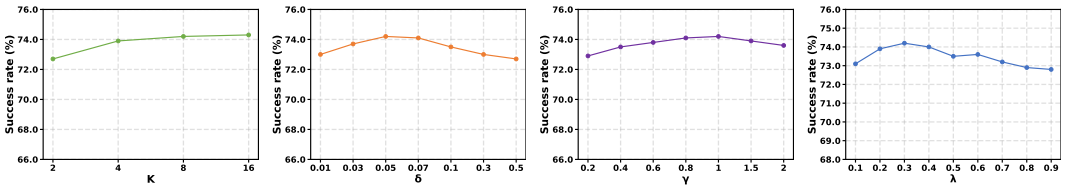
The examples compare execution with and without Future Revision under intermediate execution errors. Without Future Revision, deviations such as object slipping or dropping can propagate to subsequent actions and eventually lead to missed or incomplete placement. With Future Revision, the policy can use the predicted future information to adjust subsequent action generation and recover from these intermediate deviations in the illustrated cases. These examples illustrate how Future Revision can support recovery from intermediate execution deviations in the shown cases.

I.4 Ablation Study

Table 6 compares four variants of CAG-WAM on RoboCasa and LIBERO. Base WAM denotes the original model without additional policy optimization or Future Revision. + GRPO applies standard group-relative policy optimization, while + CAG-GRPO replaces the standard advantage with causal action gain. + Future Revision further introduces the proposed inference-time action revision on top of CAG-GRPO. The results support the effectiveness of these components in CAG-WAM.

Table 6: Ablation study. GRPO denotes standard group-relative policy optimization, CAG-GRPO uses causal action gain as the optimization advantage, and Future Revision denotes the proposed inference-time action revision.

Method	Policy Optimization	RoboCasa Avg. (%)	LIBERO Avg. (%)
Base WAM	–	71.5	98.5
+ GRPO	GRPO	71.8	98.7
+ CAG-GRPO	CAG-GRPO	73.3	99.4
+ Future Revision	CAG-GRPO	**74.2**	**99.8**



(a) K vs. Success Rate (b) δ vs. Success Rate (c) γ vs. Success Rate (d) λ vs. Success Rate

Figure 6: Parameter sensitivity.

I.5 Parameter Sensitivity

We further analyze the sensitivity of CAG-WAM to main hyperparameters, as shown in **Figure 6**.

The candidate number K controls the number of action chunks sampled from the same observation during policy optimization. We evaluate $K \in \{2, 4, 8, 16\}$. The performance improves as K increases up to 8, while further increasing the candidate number provides only a small additional gain and requires additional computation for candidate generation and paired-future evaluation. We therefore set $K = 8$ in the final configuration.

The revision threshold δ determines when Future Revision is activated according to the causal action gain of the initial action. We search $\delta \in \{0.01, 0.03, 0.05, 0.07, 0.1, 0.3, 0.5\}$, and the performance peaks at $\delta = 0.05$.

The future-feedback coefficient γ controls the contribution of the difference between the action-conditioned and environmental futures during the additional action generation. We evaluate $\gamma \in \{0.2, 0.4, 0.6, 0.8, 1.0, 1.5, 2.0\}$, with $\gamma = 1.0$ achieving the highest performance among the evaluated settings.

Finally, the action-blending coefficient λ controls the contribution of the revised action when combining it with the initial action. We vary λ from 0.1 to 0.9 with an interval of 0.1, and the results peak at $\lambda = 0.30$.

Based on these results, we use $K = 8$, $\delta = 0.05$, $\gamma = 1.0$, and $\lambda = 0.30$ as the final configuration.

I.6 Trade-off Performance

We further analyze the computational trade-off of CAG-WAM during training and inference, as shown in **Table 7**. During training, the additional computation mainly comes from paired future prediction and evaluation for policy optimization. During inference, paired future evaluation is first used to determine whether the current action requires revision, and an additional action generation is performed only when Future Revision is activated. The results show that the additional revision step introduces a moderate increase in inference latency, suggesting a favorable trade-off between performance and computational cost.

I.7 Real World Experiments

We further evaluate CAG-WAM on a real-world dual-arm robotic platform built with two NERO robotic arms from AgileX Robotics. The real-world experiments instantiate CAG-WAM with the

Table 7: Computational cost of CAG-WAM during training and inference. Inference time is measured per action-chunk decision.

Setting	Runtime	GPU Peak Memory
Training	40.8 s / update	32.3 GiB
Inference w/o Future Revision	4.82 s	8.43 GiB
Inference w/ Future Revision	5.77 s	8.43 GiB

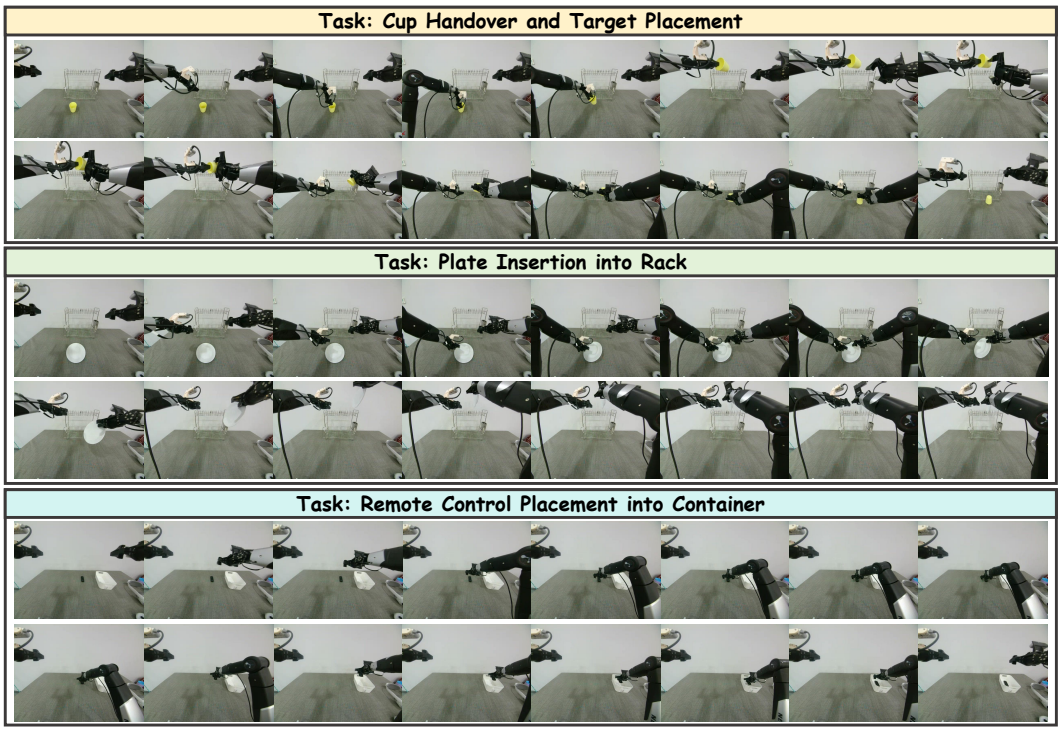


Figure 7: Real-world experiments.

observation and action spaces of the physical dual-arm platform, while retaining the same formulation for paired future evaluation, policy optimization, and action revision. The platform consists of two 7-DoF NERO robotic arms equipped with parallel grippers, and the policy operates in a closed-loop manner using visual observations and language task instructions.

We construct six tabletop manipulation tasks with different interaction requirements. *Cup Handover and Target Placement* requires the robot to transfer a cup between the two arms and place it at a designated target position. *Bottle Placement into Rack* requires grasping and placing a bottle at a designated rack position. *Toy Placement into Bowl* requires picking up a tabletop toy and placing it inside a target bowl. *Plate Insertion into Rack* requires grasping a plate, adjusting its orientation, and inserting it into a plate rack. *Drawer Opening and Closing* requires the robot to interact with a drawer and complete both opening and closing operations. *Remote Control Placement into Container* requires picking up a remote control and placing it inside a target bin. Together, these tasks cover object transfer, pick-and-place, precise placement, insertion, and articulated-object manipulation.

For each task, the robot starts from a predefined initial configuration, while the positions and orientations of task objects are varied within a predefined workspace. We evaluate each task over 10 trials and use task success rate as the primary metric. A trial is considered successful when the robot completes the instructed manipulation and reaches the corresponding target state without manual correction during execution. **Figure 7** provides representative execution examples for real-world

Table 8: Real-world evaluation results on six manipulation tasks.

Task	Success Rate (%)
Cup Handover and Target Placement	70.00
Bottle Placement into Rack	60.00
Toy Placement into Bowl	70.00
Plate Insertion into Rack	50.00
Drawer Opening and Closing	90.00
Remote Control Placement into Container	80.00
Average	70.00

tasks. **Table 8** summarizes the real-world evaluation results. The results across the manipulation tasks support the effectiveness of CAG-WAM in real-world robotic manipulation.

J Broader Impacts and Limitations

In this section, we briefly discuss the broader impacts and limitations of this work.

Broader Impacts. This work studies action evaluation in World Action Models and proposes CAG-WAM, which characterizes the future change associated with a candidate action by comparing its action-conditioned future with the corresponding environmental future. Causal action gain is used as the optimization advantage during training and as a criterion for activating Future Revision during inference, providing a way to incorporate paired future evaluation into robot policy learning and decision making. Beyond the benchmarks considered in this paper, the proposed formulation may also provide a useful perspective for studying how action-induced and environment-driven changes jointly influence future evaluation in predictive robot policies.

Limitations. Our experiments are conducted on a limited set of World Action Model backbones and robotic manipulation benchmarks. Future work may evaluate CAG-WAM with larger-scale models, a broader range of manipulation tasks, and additional robot embodiments to further study its scalability and general applicability.