THINKWORLD: WORLD IMAGINATION REINFORCES CHAIN-OF-DYNAMICS REASONING

Anonymous authors
Paper under double-blind review

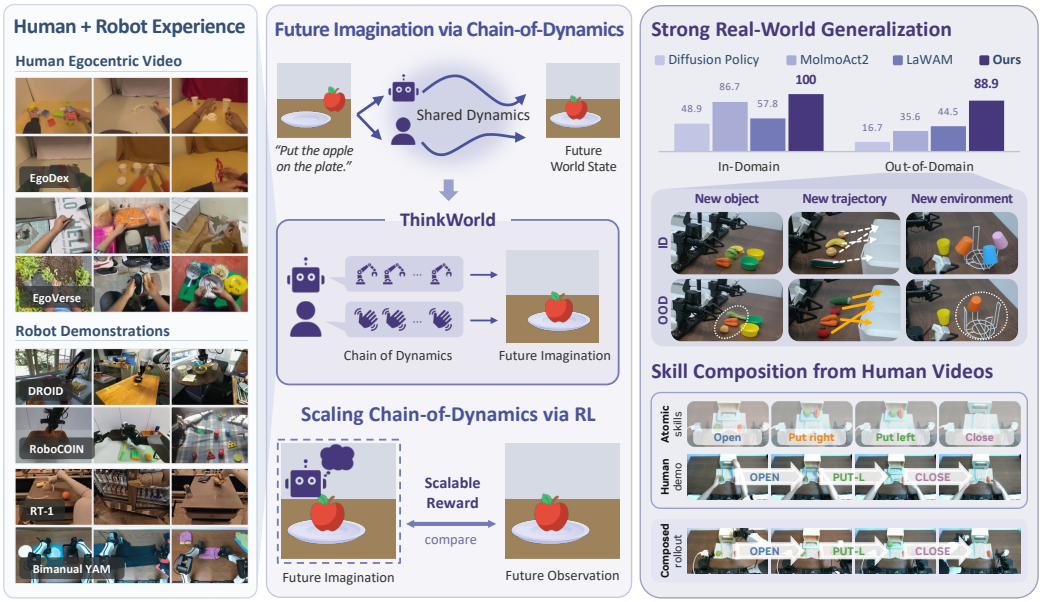


Figure 1: We introduce **ThinkWorld**, a vision-language-action framework that learns transferable Chain-of-Dynamics (CoD) reasoning from diverse human and robot experiences. CoD captures action-induced dynamics across embodiments and is reinforced through future imagination, enabling robust generalization in the real world and skill composition guided by human videos.

ABSTRACT

Vision-Language-Action (VLA) models have made substantial progress in robotic manipulation, but their generalization remains constrained by the diversity and coverage of action-labeled robot demonstrations. Human videos provide much broader behavioral and environmental experience, while learning from them faces a fundamental trade-off between scalable data and explicit action grounding. We introduce *ThinkWorld*, a VLA learning framework that grounds action reasoning in world dynamics and strengthens it through reinforcement learning with world imagination. ThinkWorld represents each action thought as a Chain-of-Dynamics (CoD), a sequence of discrete motion tokens that captures the intermediate dynamics connecting the current world to a future state and serves as a common action representation across embodiments. We first initialize CoD with action-labeled demonstrations, then predict the future induced by each CoD and reinforce those whose predictions better match the observed outcome. This enables videos without action annotations to provide direct supervision from observed state transitions, allowing human and robot experience to refine the same reasoning space. Experiments in simulation and on real robots demonstrate effective human-to-robot transfer and strong generalization to unseen tasks and environments.

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.

1 INTRODUCTION

Building general-purpose action models that transfer across tasks, objects, environments, and embodiments is a central goal of robot learning. Recent vision-language-action models (VLAs) (Brohan et al., 2023; O'Neill et al., 2024; Team et al., 2024; Kim et al., 2024; Black et al., 2024a; Intelligence et al., 2025; Bjorck et al., 2025) have made substantial progress by transferring pretrained vision-language knowledge and scaling across heterogeneous robot demonstrations. Nevertheless, their policy learning and behavioral coverage remain largely tied to action-labeled robot trajectories, whose collection requires physical platforms, embodiment-specific interfaces, and substantial human effort. Human videos offer a complementary source of experience with far greater behavioral and environmental diversity, but do not provide directly executable robot actions.

Existing approaches seek to extract actionable supervision from human videos along two main directions. Latent action models (Ye et al., 2025; Bu et al., 2025; Dai et al., 2026; Sun et al., 2026a) infer transition variables from pairs or sequences of frames and use them for representation or policy learning. While naturally scalable to action-unlabeled videos, these inferred variables are underconstrained and may capture appearance changes or camera motion rather than controllable behavior. In contrast, fully supervised approaches recover hand poses, object interactions, or retargeted trajectories to provide explicit motion supervision (Kareer et al., 2025; Zheng et al., 2026; Liu et al., 2026b). Such supervision provides direct physical grounding, but relies on accurate pose estimation and human-to-robot correspondence. Human-video learning therefore yields a trade-off between stronger action grounding from explicit supervision and broader behavioral coverage from unlabeled videos.

Future observations provide a natural supervisory signal to bridge this gap as every video records the consequences that follow a behavior, regardless of embodiment. Predictive world models (Agarwal et al., 2026; Zhou et al., 2024; Assran et al., 2025) and foresight-based VLAs (Zhang et al., 2026; Sun et al., 2026a; Cai et al., 2026; Li et al., 2026b) exploit this signal by forecasting future pixels, visual features, or trajectories, providing a direct account of *what changes* in the world. However, inferring the intermediate dynamics that result in these changes remains challenging, particularly from large-scale human videos without action annotations. Recent reasoning-augmented VLAs offer a useful perspective by introducing intermediate tokens, such as textual plans (Zawalski et al., 2024; Huang et al., 2026b), visual guidance (Lee et al., 2025; Fang et al., 2026), continuous latents (Huang et al., 2026a), or coarse action trajectories (Zhong et al., 2026), to structure action generation. This raises a natural question of whether such intermediate representations can instead be grounded in the dynamics they induce in the world, providing a shared action interface across embodiments.

Motivated by this perspective, we introduce *ThinkWorld*, a VLA learning framework that formulates action thoughts as Chain-of-Dynamics (CoD) and reinforces them through world imagination. CoD represents an action as a sequence of discrete motion tokens that captures how the world evolves over time, providing a common reasoning interface across human and robot behaviors. We first initialize this space with action-labeled demonstrations, grounding CoD in physically meaningful behavior. ThinkWorld then predicts the future world representation induced by each CoD and compares it with the future observation, using their alignment to reinforce action reasoning. As a result, videos even without motion annotations provide a learning signal, allowing large-scale human videos and robot trajectories to jointly refine the same CoD space. This training objective propagates through the shared VLM, enriching the representations for action prediction with knowledge of how motion transforms the world. Across simulation benchmarks and real-robot evaluations, we demonstrate that ThinkWorld enables human-to-robot skill composition and strong generalization to unseen scenarios.

Our contributions are as follows:

- We introduce *ThinkWorld*, a VLA learning framework that couples a common VLM-native action-reasoning interface for both human and robot with internal world imagination.
- We propose consequence-grounded reinforcement learning, which rewards CoD according to its consistency with observed future consequences and thereby enables action reasoning to improve from videos without action labels.
- We show that ThinkWorld benefits from scaling action-unlabeled human video and enables human-to-robot skill transfer, allowing robots to execute complete tasks demonstrated in human videos while receiving robot supervision only for the constituent atomic skills.

2 Related Works

Generalist Robot Policies. Vision-Language-Action (VLA) models learn generalist manipulation policies by mapping visual observations and language instructions to executable actions (Kim et al., 2024; Bjorck et al., 2025; Black et al., 2024a; Intelligence et al., 2025; Wang et al., 2026; Chen et al., 2025b; Team et al., 2025). Despite acquiring broad capabilities from large-scale robot demonstrations, end-to-end imitation remains bounded by the behaviors observed during training. Recent embodied reasoning methods introduce intermediate reasoning between perception and control (Zawalski et al., 2024; Huang et al., 2026b; Lee et al., 2025; Huang et al., 2026a; Fang et al., 2026; Zhong et al., 2026). ThinkAct (Huang et al., 2026b) reinforces textual and visual plans with action-aligned rewards, MolmoAct (Lee et al., 2025) reasons through spatial traces, Fast-ThinkAct (Huang et al., 2026a) compresses reasoning into latent thoughts, and ACoT-VLA (Zhong et al., 2026) predicts coarse reference actions to guide policy generation. However, these methods largely augment robot demonstrations with intermediate reasoning, without explicitly grounding the learned thoughts in the task-relevant changes they induce in the world. Their generalization to novel scenes and skills therefore remains constrained by the behavioral coverage of the underlying robot data.

Learning Robot Policies from Human Videos. Improving generalization also requires expanding the behavioral coverage of the training data. Human videos capture substantially greater diversity in scenes, objects, and skills than robot demonstrations, motivating their use for scaling robot learning (Yang et al., 2025; Zheng et al., 2026; Kim et al., 2026a; Li et al., 2026a). Supervised methods convert human interactions into robot-compatible representations, as in EgoScale (Zheng et al., 2026) and EgoWAM (Li et al., 2026a), but require costly cross-embodiment supervision. Latent action models instead infer action-like variables from visual transitions without explicit control labels (Ye et al., 2025; Chen et al., 2025c; Routray et al., 2026), with LAPA (Ye et al., 2025) learning discrete latent actions before grounding them into robot control. Such representations may capture appearance or embodiment cues rather than task-relevant dynamics. ThinkWorld grounds shared motion tokens and refines them by matching predicted and observed futures in videos.

World Models and Visual Foresight for Embodied Control. World models learn action-conditioned dynamics to support planning and policy optimization, with Dreamer-style agents training policies through imagined rollouts in learned environments (Hafner et al., 2023; 2025) and recent large-scale models acquiring broad physical priors from diverse video data (Agarwal et al., 2026; Gao et al., 2026). In robotics, visual-foresight methods generate video plans or image subgoals to guide control (Du et al., 2023; Black et al., 2024b; Bharadhwaj et al., 2024; Zhao et al., 2025), while predictive VLAs forecast future pixels or latent features for planning and representation learning (Zhang et al., 2026; Sun et al., 2026a; Cai et al., 2026; Li et al., 2026b). More recent world-action models jointly model future-world evolution and robot actions (Ye et al., 2026; Pai et al., 2025; Cen et al., 2025; Kim et al., 2026b). In contrast, ThinkWorld does not use future prediction as the control output or an imagined rollout for policy optimization. Instead, it treats agreement between predicted and observed futures as a training-time signal to reinforce action-grounded Chain-of-Dynamics within the VLM. This directly turns observed world changes, including those in action-unlabeled videos, into supervision for action reasoning, while future prediction is discarded at deployment.

3 Method

3.1 Problem Formulation

Given a language instruction l, a current visual observation o_t, sampled history frames $\mathbf{h}_t$, and proprioceptive state s_t in case of robot, our goal is to predict a continuous robot action sequence $\mathbf{a}_t$. Rather than treating actions solely as supervision for continuous control, ThinkWorld turns them into a reasoning interface for modeling how an action should unfold and transform the world. Through supervised fine-tuning, we instantiate this interface as a *Chain-of-Dynamics (CoD)* $\mathbf{c}_t$, grounding its discrete motion vocabulary with action-labeled demonstrations across human and robot embodiments while learning CoD-conditioned future imagination through learnable queries that predict the future visual state (Sec. 3.2). We then reinforce CoD reasoning through world imagination, using the alignment between the imagined future and the observed future $o_{t+\Delta}$ after a physical horizon Δ as a learning signal that does not require motion annotations (Sec. 3.3). Together, action grounding and reinforcement through world imagination refine the model representations used for continuous control while extending learning to action-unlabeled videos at scale.

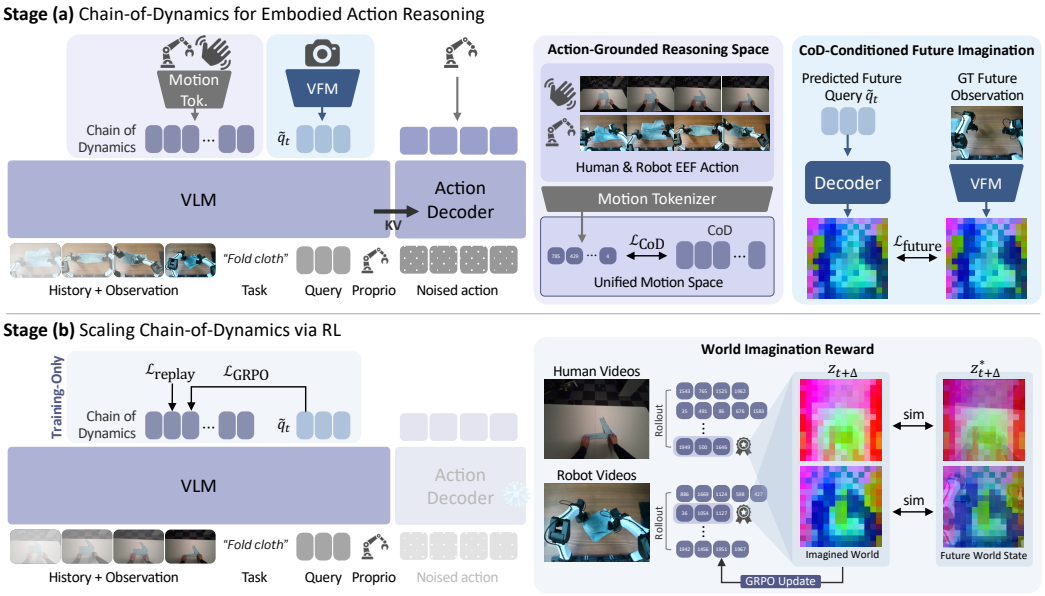


Figure 2: **ThinkWorld architecture and training stages.** In stage (a), supervised fine-tuning (SFT) tokenizes robot actions and annotated human hand motions into a unified discrete motion space to ground Chain-of-Dynamics (CoD) reasoning, while CoD-conditioned imagination predicts the future visual feature. In stage (b), reinforcement learning (RL) uses the alignment between imagined and observed futures as a scalable reward for CoD reasoning over human and robot videos, strengthening action reasoning without motion annotations and improving guidance for continuous control.

3.2 Chain-of-Dynamics for Embodied Action Reasoning

Robot actions are typically represented in embodiment-specific control spaces, such as joint positions or end-effector poses. Although directly executable, these control targets keep action supervision tied to individual embodiments, limiting transfer across robots and from human demonstrations. We therefore introduce *Chain-of-Dynamics (CoD)*, a sequence of discrete motion tokens that provides a shared action language for human and robot motion, as illustrated in Fig. 2(a).

Action-Grounded Reasoning Space. Let $\mathbf{m}_t$ denote a motion segment from a robot trajectory or an annotated human hand trajectory. We map both to a normalized delta end-effector representation and tokenize them with a FAST-style motion tokenizer $\mathcal{T}(\cdot)$ (Pertsch et al., 2025), obtaining the target CoD sequence $\mathbf{c}_t^\star = \mathcal{T}(\mathbf{m}_t)$ in a shared discrete vocabulary. Conditioned on the instruction, visual history, and current observation, the VLM learns to generate $\mathbf{c}_t$ autoregressively. This tokenization provides a common interface grounded in motion from both sources without assuming kinematic equivalence between humans and robots.

CoD-Conditioned Future Imagination. Motion grounding defines a CoD sequence, but not the world change it induces. To predict this change, we append learnable future queries $\mathbf{q}_{t+\Delta}$ after the CoD. Given the visual context and teacher-forced CoD $\mathbf{c}_t^\star$, the VLM contextualizes these queries as $\tilde{\mathbf{q}}_{t+\Delta}$, which a lightweight decoder $\mathcal{D}_{\text{future}}$ maps to the predicted future representation $\mathbf{z}_{t+\Delta}$ at physical horizon Δ:

$$\begin{aligned} \tilde{\mathbf{q}}_{t+\Delta} &= \text{VLM}(l, \mathbf{h}_t, o_t, \mathbf{c}_t^\star, \mathbf{q}_{t+\Delta}), \\ \mathbf{z}_{t+\Delta} &= \mathcal{D}_{\text{future}}(\tilde{\mathbf{q}}_{t+\Delta}). \end{aligned} \tag{1}$$

Instead of reconstructing future RGB pixels, we predict in a frozen DINO feature space. DINO features retain dense spatial and object structure while abstracting away appearance details (Zhou et al., 2024; Siméoni et al., 2025), allowing imagined and observed futures to be compared by structural change without photorealistic generation. The target $\mathbf{z}_{t+\Delta}^\star$ is obtained by encoding the observed future frame $o_{t+\Delta}$ with the frozen DINO encoder $\mathcal{E}_{\text{future}}$. This frame is used only to construct the target and is never provided as VLM input.

Joint Supervision of CoD and Future Prediction. We jointly train the VLM to generate CoDs from observed motion and predict future visual states conditioned on those CoDs. The objective is

$$\mathcal{L}_{\text{SFT}} = \mathcal{L}_{\text{CoD}} + \lambda_{\text{future}}\mathcal{L}_{\text{future}},$$
$$\text{where} \quad \mathcal{L}_{\text{CoD}} = \text{CE}(p_\theta(\mathbf{c}_t \mid l, \mathbf{h}_t, o_t), \mathbf{c}_t^\star), \quad \mathcal{L}_{\text{future}} = \text{dist}\big(\mathbf{z}_{t+\Delta}, \mathbf{z}_{t+\Delta}^\star\big). \tag{2}$$

Here, $\text{dist}(\cdot,\cdot)$ is a feature distance metric and λ_{future} weights the future prediction loss. For continuous control, the action expert uses the VLM representations shaped by these objectives, rather than decoding or directly consuming CoD tokens (Intelligence et al., 2025; Chen et al., 2025a; Sun et al., 2026b). Unlike CoD targets, future prediction targets are available from *any* temporally ordered human or robot video. In Sec. 3.3, we use observed futures to evaluate sampled CoDs and extend learning to transitions without motion annotations.

3.3 Scaling Chain-of-Dynamics Reasoning via Reinforcing World Imagination

Once trained with joint supervision, the model generates CoDs and predicts future visual states conditioned on them. However, prediction under teacher-forced CoDs alone does not train the VLM to plan the intermediate dynamics needed to reach a desired future state. Therefore, we reward sampled CoDs according to the agreement between their imagined and observed futures, encouraging the VLM to reason about the world changes implied by its CoDs, as shown in Fig. 2(b). Since this reward requires only observed transitions, it enables the VLM to explore alternative CoDs and learn from human and robot videos without requiring action labels or executing candidate motions.

Reward Shaping from World Imagination. For each candidate CoD, ThinkWorld predicts its future visual representation and compares it with the observed future in latent space. We compute the reward using a *motion-enhanced feature similarity* score $\text{sim}(\cdot,\cdot)$, which places greater weight on DINO feature agreement in patches with larger current-to-future changes:

$$r_{\text{world}} = \text{sim}(\mathbf{z}_{t+\Delta}, \mathbf{z}_{t+\Delta}^\star). \tag{3}$$

The score retains a uniform contribution from the full scene to preserve global consistency while preventing redundant background regions from dominating the reward (Shah et al., 2026). Motion weighting uses only observed frames, so scoring requires no action annotations.

Optimizing CoD Reasoning with World-Imagination RL. We aim to learn CoDs that capture the intermediate dynamics leading from the current scene to a desired future state. To train this reasoning process, ThinkWorld samples alternative CoDs from the same visual context and rewards those whose imagined futures best match the observed future. These rewards let ThinkWorld explore alternative CoDs that could lead from the current scene to the observed future, allowing passive human and robot videos to guide action reasoning without action labels or online execution of candidate motions.

We update the VLM with GRPO (Shao et al., 2024) using normalized world-imagination rewards (Liu et al., 2026a), while supervised replay preserves the motion grounding learned during SFT:

$$\mathcal{L}_{\text{RL}} = \lambda_{\text{GRPO}}\mathcal{L}_{\text{GRPO}}(r_{\text{world}}) + \lambda_{\text{replay}}\mathcal{L}_{\text{replay}}. \tag{4}$$

By comparing imagined and observed futures, this stage encourages the VLM to encode action-relevant dynamics in the representations later used for continuous control. All other components remain frozen during RL, and neither CoD generation nor future prediction is required at deployment. Complete optimization details are provided in Appendix A.6.

4 Experiment

4.1 Implementation Details

Training Setup. For the motion tokenizer $\mathcal{T}(\cdot)$, we convert both robot actions and human wrist poses into a unified delta end-effector representation. We use Qwen3-VL-2B (Bai et al., 2025) as the VLM backbone with the action expert from MolmoAct2 (Fang et al., 2026). Future queries are decoded by a Transformer decoder, with ground-truth future representations extracted using DINOv3 (Siméoni et al., 2025). Our training proceeds in two stages. During *SFT and RL pre-training*, we first fully fine-tune the VLM for CoD reasoning and future prediction, and then adapt the action expert while freezing the VLM following Driess et al. (2026). We subsequently perform RL with

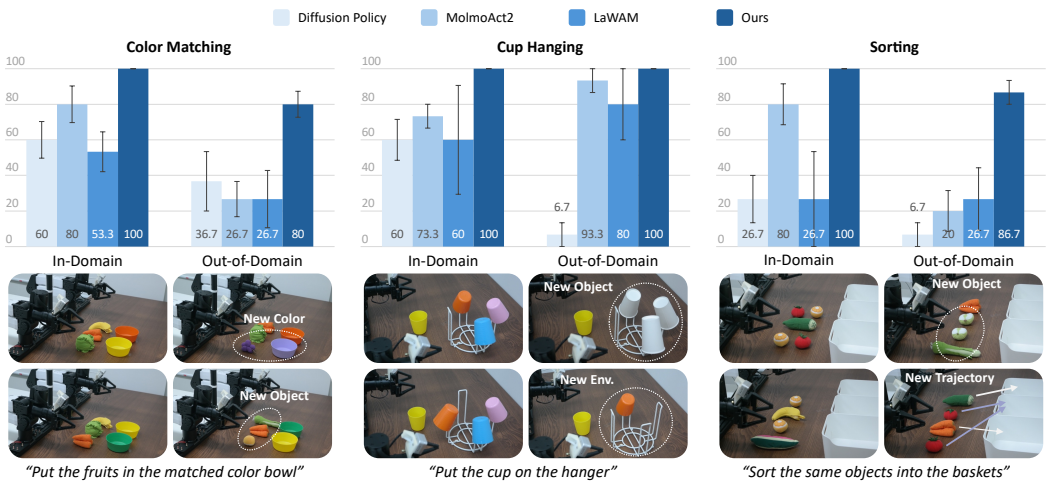


Figure 3: **Real-world out-of-distribution (OOD) generalization.** Success rates for color matching, cup hanging, and sorting under in-domain and OOD conditions. OOD settings introduce new colors, objects, environments, or sorting trajectories, as illustrated below the plots. ThinkWorld achieves the highest OOD success rates across all three tasks, despite substantial drops for the baselines.

$K = 4$ sampled trajectories, updating only the VLM while keeping the action expert frozen. We use 128 and 64 NVIDIA H100 GPUs for SFT and RL, respectively. During *Embodiment-Specific Post-Training*, we further adapt the pretrained model to each target embodiment. ThinkWorld-LIBERO is trained on the standard LIBERO training set to evaluate generalization and robustness to novel environments and perturbations, while ThinkWorld-YAM is adapted on MolmoAct2-BimanualYAM for real-robot experiments. We use 32 and 64 NVIDIA H100 GPUs for the two settings, respectively.

Training Datasets. Our training corpus combines human egocentric data from EgoDex (Hoque et al., 2026) and EgoVerse (Punamiya et al., 2026) with robot demonstrations from MolmoAct2-DROID/BimanualYAM (Fang et al., 2026), Bridge/RT-1 (O'Neill et al., 2024), BC-Z (Jang et al., 2021), and RoboCOIN (Wu et al., 2025). During SFT, we also use auxiliary VQA data to preserve visual-language capabilities and reduce catastrophic forgetting. Wrist views are included when available; otherwise, only the main view is used.

Real-Robot Experimental Setup. We use a bimanual YAM system equipped with three cameras for real-robot evaluation: one global top-view camera and one wrist-view camera for each arm. We use only RGB observations for both training and inference. The embodiment-adapted ThinkWorld-YAM model is then used as the initialization for efficient task-specific fine-tuning. Additional implementation and training details are provided in Appendix.

4.2 Real-World Generalization under Distribution Shifts

We evaluate real-world generalization under matched in-domain (ID) and out-of-distribution (OOD) settings across three manipulation tasks: placing cups on a hanger, color matching, and object sorting. OOD settings introduce changes in object color or category and, in some cases, require trajectories not observed during task-specific training. For each task, we collect demonstrations across at least six object configurations and evaluate multiple settings under both ID and OOD conditions, with five trials per setting. These tasks require the policy to identify task-relevant object attributes, such as color and category, and adapt its motion accordingly.

As shown in Fig. 3, we compare ThinkWorld-YAM with Diffusion Policy (Chi et al., 2023), MolmoAct2 (Fang et al., 2026), a VLA with explicit action reasoning, and LaWAM (Chen et al., 2026), which conditions action generation on predicted latent visual subgoals. While the methods perform similarly in-domain, their differences become more pronounced under OOD shifts. ThinkWorld maintains higher success rates when object appearances change or unseen trajectories are required, outperforming both MolmoAct2 and LaWAM. This suggests that grounding CoD reasoning in action-induced world changes helps the policy adapt to task-relevant changes rather than relying on appearance cues or memorized trajectories.

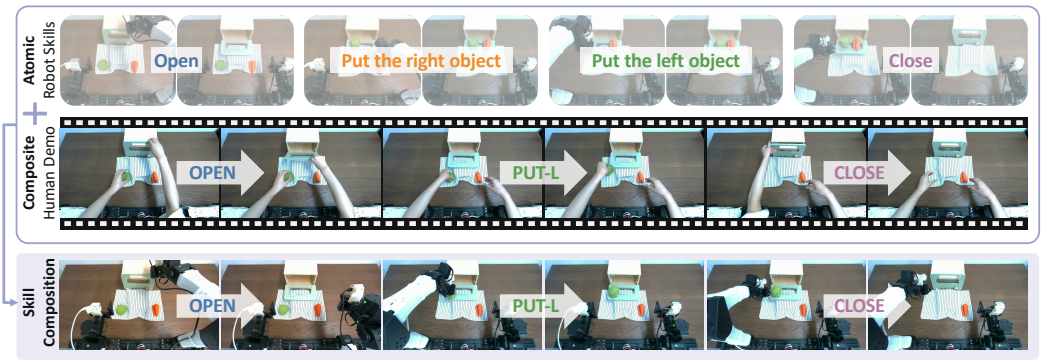


Figure 4: **Composing atomic robot skills from a human demonstration.** The *top* shows atomic robot skills, while the *middle* shows a human demonstration that provides the complete task composition, e.g., open → put-left → close. Guided by this human demonstration, ThinkWorld composes the corresponding robot skills and executes the complete task in a single rollout at the *bottom*.

Table 1: **Success rates (%) on LIBERO and LIBERO-PRO.** LIBERO-PRO axis values are arithmetic means over Goal, Spatial, LIBERO-10, and Object. LIBERO-PRO baseline results are taken from the official repository unless marked with †, which denotes our reproduced result. Best and second-best results within each benchmark column are shown in **bold** and underlined, respectively.

		LIBERO					LIBERO-PRO					
Model	Params.	Spatial	Object	Goal	Long	Avg.	Object	Position	Semantic	Task	Environment	Avg.
OpenVLA (Kim et al., 2024)	7.5	84.7	88.4	79.2	53.7	76.5	93.0	0.0	**97.3**	0.0	**68.0**	51.7
π_0 (Black et al., 2024a)	3.2	96.8	98.8	95.8	85.2	94.2	90.5	0.0	90.5	0.0	38.8	44.0
$\pi_{0.5}$ (Intelligence et al., 2025)	3.4	**98.8**	98.2	**98.0**	92.4	96.9	**96.0**	**20.8**	95.8	0.8	52.8	53.2
MolmoAct (Lee et al., 2025)	8.1	87.0	95.4	87.6	77.2	86.6	76.0	1.5	85.8	1.5	–	41.2
VLA-JEPA (Sun et al., 2026a)	2.8	96.2	**99.6**	97.2	**95.8**	97.2	80.4	3.5	97.2	20.8	41.0	48.5†
ThinkWorld (Ours)	2.9	98.2	98.8	97.8	94.2	**97.3**	73.3	18.2	93.9	**31.9**	60.9	**55.6**

4.3 Acquiring Skill Composition from Human Experience

We further evaluate whether human demonstrations can transfer compositional knowledge to robot control. We construct an oven manipulation task from reusable atomic robot skills and provide robot demonstrations only for the individual skills *1) open, 2) close, 3) put the left object in the oven*, and *4) put the right object in the oven*, while a human video demonstrates how a complete compositional task can be achieved. Existing end-to-end VLA approaches typically learn executable control from robot action supervision (Fang et al., 2026), making it difficult to compose complete robot tasks from human demonstrations when only atomic robot skills are supervised. Successful execution requires the robot to identify the relevant skills, preserve their temporal order, and transition between them without observing the complete robot trajectory during training. As shown in Fig. 4, a human demonstration of *open* → *put-left* → *close* enables ThinkWorld to compose the corresponding robot skills and execute the full sequence in a single rollout. These results demonstrate that the shared CoD reasoning space enables compositional knowledge to transfer from human demonstrations to robot execution, allowing the robot to perform a complete skill sequence never demonstrated in robot data.

4.4 Robustness to Task and Environment Shifts in Simulation

We evaluate ThinkWorld on standard LIBERO (Liu et al., 2023) and the more challenging LIBERO-PRO (Zhou et al., 2025). LIBERO evaluates manipulation across spatial, object, goal, and long-horizon task suites, while LIBERO-PRO introduces controlled shifts in object attributes, positions, instruction semantics, task goals, and environments. ThinkWorld-LIBERO is post-trained only on the standard LIBERO training set and evaluated directly on LIBERO-PRO without exposure to these perturbations. As shown in Tab. 1, ThinkWorld achieves the best overall performance on both benchmarks, with average success rates of 97.3% on LIBERO and 55.6% on LIBERO-PRO.

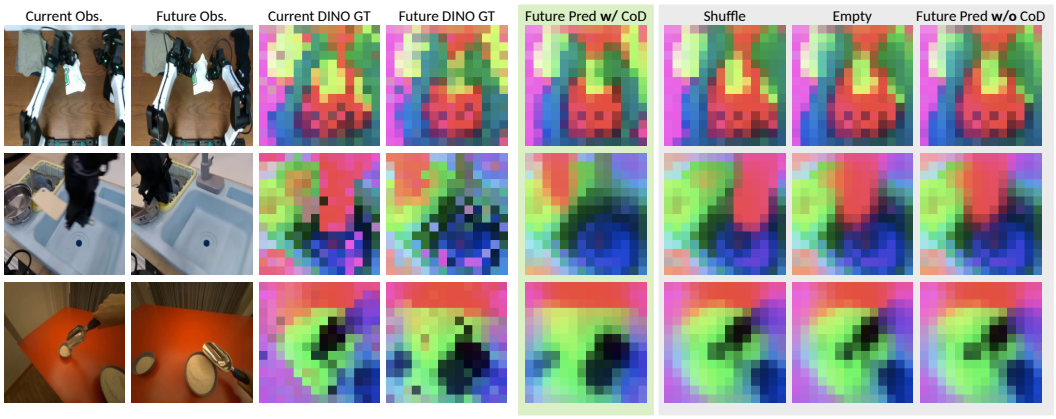


Figure 5: **Effect of Chain-of-Dynamics (CoD) on future prediction.** We compare future DINO predictions under generated, shuffled, empty, and no-CoD conditions against the current and future ground-truth features. The generated CoD captures the future-state structure, whereas the other variants largely remain similar to the current state and fail to reflect the expected future change.

Notably, under task and environment shifts, ThinkWorld reaches 31.9% and 60.9%, outperforming VLA-JEPA by 11.1 and 19.9 percentage points, respectively. While VLA-JEPA also incorporates latent world modeling, these gains suggest that grounding CoD reasoning in action-induced world changes provides more transferable structure for adapting to shifts beyond the training distribution.

4.5 Analysis

Cross-Embodiment Structure of CoD. We further examine whether CoD induces a shared motion-centric structure across human and robot embodiments. For each sample, we extract the VLM hidden state of the final `</think>` token, which summarizes the generated CoD, and visualize these features with t-SNE in Fig. 6. We use bimanual human and robot datasets, with colors denoting four 2D end-effector motion directions and marker shapes indicating dataset sources. Despite substantial differences in embodiment and visual appearance, samples with similar motion directions form coherent clusters that contain examples from multiple datasets, while samples from the same dataset separate according to their motions. This suggests that CoD features are organized primarily around action-relevant motion rather than embodiment identity, revealing a shared action-level structure shared across human and robot behaviors.

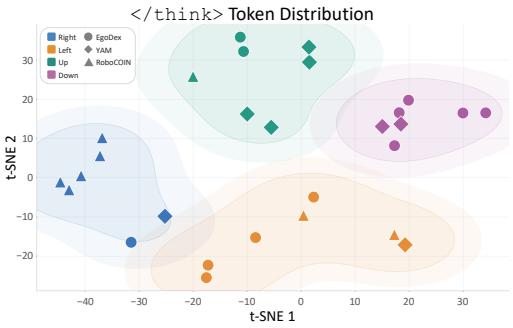


Figure 6: **t-SNE visualization of CoD across embodiments.** Colors denote four end-effector motion directions and marker shapes denote datasets. Similar motions cluster across human and robot data rather than by source, suggesting action structure shared across embodiments.

Role of CoD in Future Prediction. We first examine how CoD contributes to future prediction. As shown in Fig. 5, predictions conditioned on the generated CoD better capture the action-induced transition toward the ground-truth future, whereas shuffled, empty, or no-CoD variants tend to preserve the current-state structure or predict incorrect transitions. Tab. 2 quantitatively supports this observation: CoD achieves a cosine similarity of 0.74 across all patches, compared with 0.71–0.72 for the other conditions. The difference is more pronounced in the top 25% most dynamic target patches, where CoD reaches 0.62 compared with 0.54–0.55, indicating that CoD is particularly effective at predicting action-induced changes.

We further examine what information CoD captures through attention visualization in Fig. 7. Across both human and robot examples, CoD consistently attends to action-relevant dynamic regions,

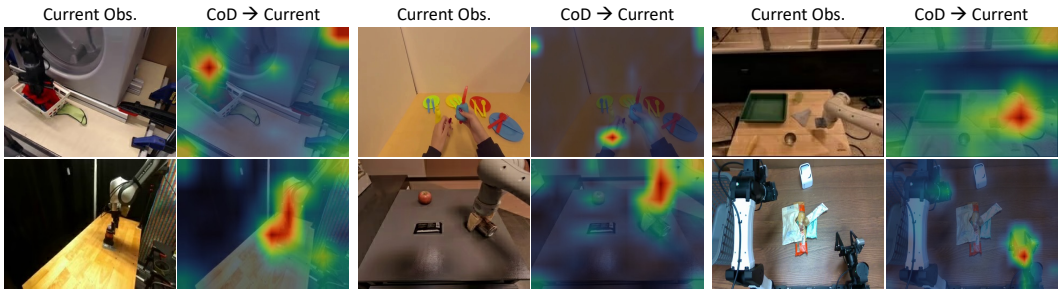


Figure 7: **Attention analysis of CoD.** We visualize attention from CoD tokens to the current observation. Across human and robot examples, CoD attends to action-relevant regions, including human hands, robot arms, and manipulated objects, highlighting its focus on motion-relevant cues.

Table 2: **Effect of CoD on future prediction.** Cosine similarity to ground-truth DINOv3 features one second ahead, over all patches and top-25% motion patches. Δ is relative to CoD on; higher is better. Best in **bold**.

Condition	All patches		Top-25% motion	
	Cos ↑	Δ	Cos ↑	Δ
Shuffle	0.71	−0.024	0.55	−0.071
Empty	0.72	−0.020	0.54	−0.079
CoD off	0.72	−0.018	0.55	−0.071
CoD on	**0.74**	—	**0.62**	—

Table 3: **Effectiveness of training components.** Success rates (%) on LIBERO-PRO when visual history frames, future prediction objective, CoD, and RL are cumulatively added to the Qwen baseline.

Model variant	LIBERO-PRO					
	Object	Pos.	Sem.	Task	Env.	Avg.
Qwen baseline	59.5	1.9	84.4	17.2	37.8	40.1
+ History frames	61.3	4.9	84.6	22.8	29.8	40.7
+ Future prediction	70.0	11.9	89.4	26.3	54.4	50.4
+ CoD	74.5	16.9	94.4	29.7	56.4	54.4
+ RL	73.3	18.2	93.9	31.9	60.9	**55.6**

including human hands or robot arms and the manipulated objects, rather than static background regions. This pattern indicates that CoD focuses on the parts of the scene most involved in the ongoing motion and links these motion cues to the resulting changes in object state and configuration. Together with the future-prediction results, these visualizations suggest that CoD captures action-centric dynamics that ground future prediction in the state transitions induced by an action.

Effect of Training Components. To assess the effectiveness of each training component, we start from a baseline consisting of the Qwen3-VL backbone with the MolmoAct2 action expert, and progressively add visual history, future prediction, CoD, and world-imagination RL. As shown in Tab. 3, history provides temporal context and slightly improves the average success rate from 40.1% to 40.7%. Future prediction brings a much larger gain to 50.4%, indicating the benefit of explicitly modeling how the scene evolves. Adding CoD further raises performance to 54.4%, with consistent improvements across all distribution shifts, showing the value of grounding future prediction in action-relevant dynamics. Finally, world-imagination RL improves the average to 55.6%, with further gains under position, task, and environment shifts, demonstrating its role in refining CoD reasoning beyond supervised training. Overall, these ablations support the design of ThinkWorld, showing that each component contributes to more robust and generalizable action reasoning.

5 Conclusion

We present *ThinkWorld*, a VLA learning framework that learns transferable Chain-of-Dynamics (CoD) through reinforcement learning with world imagination. By grounding action thoughts in the world changes they induce, ThinkWorld learns a common representation across human and robot embodiments and extends action reasoning to videos without explicit action annotations. Extensive experiments demonstrate effective human-to-robot transfer and strong generalization to unseen tasks and environments, validating the effectiveness of learning transferable action reasoning from observed world dynamics and its potential for scaling robot learning with diverse human and robot experience.

AI USE STATEMENT

We used generative AI tools to assist with manuscript drafting and editing for clarity and readability, code editing during implementation, and discussion and feedback during the research process. Research ideas, methodological choices, experimental designs, and interpretation of results were developed and ultimately determined by the authors; generative AI was not used to independently make these scientific decisions or generate experimental results. All AI-assisted text and code were reviewed by the authors, and code changes were tested for correctness. We take responsibility for the final content of this work, including text, claims, code, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies robot learning from human and robot demonstrations for manipulation tasks in controlled research environments. Human-video data are drawn from publicly available datasets and additional demonstrations collected for research purposes. Our system is intended for research purposes and is not designed for safety-critical deployment. As unintended behaviors may occur during robot execution, real-robot experiments are conducted in controlled settings, and deployment in broader real-world environments would require additional safety validation.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of the model architecture, training objectives, and learning procedure in Sec. 3, together with experimental and implementation details in Sec. 4 and the appendix. Additional information relevant to reproducibility is provided in Appendix D and elsewhere in the appendix.

REFERENCES

Niket Agarwal, Arslan Ali, Jon Allen, Martin Antolini, Adeline Aubame, Alisson Azzolini, Junjie Bai, Maciej Bala, Yogesh Balaji, Josh Bapst, et al. Cosmos 3: Omnimodal world models for physical ai. *arXiv preprint arXiv:2606.02800*, 2026.

Mahmoud Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Homanga Bharadhwaj, Debidatta Dwibedi, Abhinav Gupta, Shubham Tulsiani, Carl Doersch, Ted Xiao, Dhruv Shah, Fei Xia, Dorsa Sadigh, and Sean Kirmani. Gen2act: Human video generation in novel scenarios enables generalizable robot manipulation. *arXiv preprint arXiv:2409.16283*, 2024.

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.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024a.

Kevin Black, Mitsuhiko Nakamoto, Pranav Atreya, Homer Walke, Chelsea Finn, Aviral Kumar, and Sergey Levine. Zero-shot robotic manipulation with pre-trained image-editing diffusion models. In *International Conference on Learning Representations*, volume 2024, pp. 33431–33452, 2024b.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

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.

Remi Cadene, Simon Alibert, Alexander Soare, Quentin Gallouedec, Adil Zouitine, Steven Palma, Pepijn Kooijmans, Michel Aractingi, Mustafa Shukor, Dana Aubakirova, Martino Russi, Francesco Capuano, Caroline Pascal, Jade Choghari, Khalil Meftah, Maxime Ellerbach, Jess Moss, and Thomas Wolf. Lerobot: State-of-the-art machine learning for real-world robotics in pytorch. `https://github.com/huggingface/lerobot`, 2024.

Junhao Cai, Zetao Cai, Jiafei Cao, Yilun Chen, Zeyu He, Lei Jiang, Hang Li, Hengjie Li, Yang Li, Yufei Liu, et al. Internvla-a1: Unifying understanding, generation and action for robotic manipulation. *arXiv preprint arXiv:2601.02456*, 2026.

Jun Cen, Chaohui Yu, Hangjie Yuan, Yuming Jiang, Siteng Huang, Jiayan Guo, Xin Li, Yibing Song, Hao Luo, Fan Wang, et al. Worldvla: Towards autoregressive action world model. *arXiv preprint arXiv:2506.21539*, 2025.

Jialei Chen, Kai Wang, Kang Chen, Shuaihang Chen, Feng Gao, Wenhao Tang, Zhiyuan Li, Weilin Liu, Zhuyu Yao, Boxun Li, et al. Lawam: Latent world action models for efficient dynamics-aware robot policies. *arXiv preprint arXiv:2606.15768*, 2026.

William Chen, Suneel Belkhale, Suvir Mirchandani, Oier Mees, Danny Driess, Karl Pertsch, and Sergey Levine. Training strategies for efficient embodied reasoning. *arXiv preprint arXiv:2505.08243*, 2025a.

Xinyi Chen, Yilun Chen, Yanwei Fu, Ning Gao, Jiaya Jia, Weiyang Jin, Hao Li, Yao Mu, Jiangmiao Pang, Yu Qiao, et al. Internvla-m1: A spatially guided vision-language-action framework for generalist robot policy. *arXiv preprint arXiv:2510.13778*, 2025b.

Yi Chen, Yuying Ge, Weiliang Tang, Yizhuo Li, Yixiao Ge, Mingyu Ding, Ying Shan, and Xihui Liu. Moto: Latent motion token as the bridging language for learning robot manipulation from videos. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 19752–19763. IEEE, 2025c.

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*, pp. 02783649241273668, 2023.

Christopher Clark, Jieyu Zhang, Zixian Ma, Jae Sung Park, Rohun Tripathi, Sangho Lee, Mohammadreza Salehi, Jason Ren, Chris Dongjoo Kim, Yinuo Yang, et al. Molmo2: Open weights and data for vision-language models with video understanding and grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 28652–28668, 2026.

Weisheng Dai, Kai Lan, Jianyi Zhou, Bo Zhao, Xiu Su, Junwen Tong, Weili Guan, and Shuo Yang. Conla: Contrastive latent action learning from human videos for robotic manipulation. *arXiv preprint arXiv:2602.00557*, 2026.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 91–104. IEEE, 2025.

Danny Driess, Jost Springenberg, Brian Ichter, Lili Yu, Adrian Li-Bell, Karl Pertsch, Allen Ren, Homer Walke, Quan Vuong, Lucy Xiaoyang Shi, et al. Knowledge insulating vision-language-action models: Train fast, run fast, generalize better. *Advances in Neural Information Processing Systems*, 38:102867–102888, 2026.

Yilun Du, Sherry Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Josh Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. *Advances in neural information processing systems*, 36:9156–9172, 2023.

Haoquan Fang, Jiafei Duan, Donovan Clay, Sam Wang, Shuo Liu, Weikai Huang, Xiang Fan, Wei-Chuan Tsai, Shirui Chen, Yi Ru Wang, et al. Molmoact2: Action reasoning models for real-world deployment. *arXiv preprint arXiv:2605.02881*, 2026.

Senyu Fei, Siyin Wang, Junhao Shi, Zihao Dai, Jikun Cai, Pengfang Qian, Li Ji, Xinzhe He, Shiduo Zhang, Zhaoye Fei, Jinlan Fu, Jingjing Gong, and Xipeng Qiu. Libero-plus: A progressive robustness benchmark for visual-language-action models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 38574–38583, June 2026.

Shenyuan Gao, William Liang, Kaiyuan Zheng, Ayaan Malik, Seonghyeon Ye, Sihyun Yu, Wei-Cheng Tseng, Yuzhu Dong, Kaichun Mo, Chen-Hsuan Lin, et al. Dreamdojo: A generalist robot world model from large-scale human videos. *arXiv preprint arXiv:2602.06949*, 2026.

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

Danijar Hafner, Wilson Yan, and Timothy Lillicrap. Training agents inside of scalable world models. *arXiv preprint arXiv:2509.24527*, 2025.

Ryan Hoque, Peide Huang, David Yoon, Jian Zhang, et al. Egodex: Learning dexterous manipulation from large-scale egocentric video. In *International Conference on Learning Representations*, volume 2026, pp. 4218–4237, 2026.

Chi-Pin Huang, Yunze Man, Zhiding Yu, Min-Hung Chen, Jan Kautz, Yu-Chiang Frank Wang, and Fu-En Yang. Fast-thinkact: Efficient vision-language-action reasoning via verbalizable latent planning. *arXiv preprint arXiv:2601.09708*, 2026a.

Chi-Pin Huang, Yueh-Hua Wu, Min-Hung Chen, Frank Wang, and Fu-En Yang. Thinkact: Vision-language-action reasoning via reinforced visual latent planning. *Advances in Neural Information Processing Systems*, 38:82782–82802, 2026b.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. π0. 5: a vision-language-action model with open-world generalization, 2025. *URL https://arxiv. org/abs/2504.16054*, 1(2):3, 2025.

Eric Jang, Alex Irpan, Mohi Khansari, Daniel Kappler, Frederik Ebert, Corey Lynch, Sergey Levine, and Chelsea Finn. BC-z: Zero-shot task generalization with robotic imitation learning. In *5th Annual Conference on Robot Learning*, 2021. URL https://openreview.net/forum?id=8kbp23tSGYv.

Simar Kareer, Dhruv Patel, Ryan Punamiya, Pranay Mathur, Shuo Cheng, Chen Wang, Judy Hoffman, and Danfei Xu. Egomimic: Scaling imitation learning via egocentric video. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 13226–13233. IEEE, 2025.

Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

Ji Woong Kim, Ke Wang, Zipeng Fu, Sirui Chen, Cong Zhao, Jeff Lai, and Chelsea Finn. Ego-pi: Vla fine-tuning for ego-centric human and robot data. *arXiv preprint arXiv:2606.08107*, 2026a.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. 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*, 2026b.

Jason Lee, Jiafei Duan, Haoquan Fang, Yuquan Deng, Shuo Liu, Boyang Li, Bohan Fang, Jieyu Zhang, Yi Ru Wang, Sangho Lee, et al. Molmoact: Action reasoning models that can reason in space. *arXiv preprint arXiv:2508.07917*, 2025.

Baoyu Li, Xinchen Yin, Mengying Lin, Yixin Zhang, and Danfei Xu. Egowam: World action models beyond pixels with in-the-wild egocentric human data. In *Robot World Models*, 2026a.

Wei Li, Peijin Jia, Yuan Ma, Xuefeng Jiang, Titong Jiang, Sheng Sun, Yujian Li, Xin Wen, Han Hong, Zhikang Liu, et al. Fomovla: Bridging visual foresight and motion guidance for vision-language-action models. *arXiv preprint arXiv:2607.14739*, 2026b.

Xuanlin Li, Kyle Hsu, Jiayuan Gu, Karl Pertsch, Oier Mees, Homer Rich Walke, Chuyuan Fu, Ishikaa Lunawat, Isabel Sieh, Sean Kirmani, et al. Evaluating real-world robot manipulation policies in simulation. *arXiv preprint arXiv:2405.05941*, 2024.

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.

Shih-Yang Liu, Xin Dong, Ximing Lu, Shizhe Diao, Peter Belcak, Mingjie Liu, Min-Hung Chen, Hongxu Yin, Yu-Chiang Frank Wang, Kwang-Ting Cheng, et al. Gdpo: Group reward-decoupled normalization policy optimization for multi-reward rl optimization. *arXiv preprint arXiv:2601.05242*, 2026a.

Yangcen Liu, Shuo Cheng, Xinchen Yin, Woo Chul Shin, Alfred Cueva, Yiran Yang, Zhenyang Chen, Chuye Zhang, and Danfei Xu. Egoengine: From egocentric human videos to high-fidelity dexterous robot demonstrations. *arXiv preprint arXiv:2606.12604*, 2026b.

Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6892–6903. IEEE, 2024.

Jonas Pai, Liam Achenbach, Victoriano Montesinos, Benedek Forrai, Oier Mees, and Elvis Nava. mimic-video: Video-action models for generalizable robot control beyond vlas. *arXiv preprint arXiv:2512.15692*, 2025.

Karl Pertsch, Kyle Stachowicz, Brian Ichter, Danny Driess, Suraj Nair, Quan Vuong, Oier Mees, Chelsea Finn, and Sergey Levine. Fast: Efficient action tokenization for vision-language-action models. *arXiv preprint arXiv:2501.09747*, 2025.

Ryan Punamiya, Simar Kareer, Zeyi Liu, Josh Citron, Ri-Zhao Qiu, Xiongyi Cai, Alexey Gavryushin, Jiaqi Chen, Davide Liconti, Lawrence Y Zhu, et al. Egoverse: An egocentric human dataset for robot learning from around the world. *arXiv preprint arXiv:2604.07607*, 2026.

Sandeep Kumar Routray, Hengkai Pan, Unnat Jain, Shikhar Bahl, and Deepak Pathak. Vipra: Video prediction for robot actions. In *International Conference on Learning Representations*, volume 2026, pp. 68823–68857, 2026.

Pierre Sermanet, Tianli Ding, Jeffrey Zhao, Fei Xia, Debidatta Dwibedi, Keerthana Gopalakrishnan, Christine Chan, Gabriel Dulac-Arnold, Sharath Maddineni, Nikhil J Joshi, Pete Florence, Wei Han, Robert Baruch, Yao Lu, Suvir Mirchandani, Peng Xu, Pannag Sanketi, Karol Hausman, Izhak Shafran, Brian Ichter, and Yuan Cao. Robovqa: Multimodal long-horizon reasoning for robotics. In *arXiv preprint arXiv:2311.00899*, 2023.

Ketul Shah, Robert Crandall, Jie Xu, Peng Zhou, Vipin Pillai, Marian George, Mayank Bansal, and Rama Chellappa. Mv2mae: Self-supervised video pre-training with motion-aware multi-view masked autoencoders. *Transactions on Machine Learning Research*, 2026.

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

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Jingwen Sun, Wenyao Zhang, Zekun Qi, Shaojie Ren, Zezhi Liu, Hanxin Zhu, Guangzhong Sun, Xin Jin, and Zhibo Chen. Vla-jepa: Enhancing vision-language-action model with latent world model. In *European Conference on Computer Vision*, pp. 478–497. Springer, 2026a.

Nan Sun, Yuan Zhang, Yongkun Yang, Wentao Zhao, Peiyan Li, Jun Guo, Wenxuan Song, Pengxiang Ding, Runze Suo, Yifei Su, et al. Revisiting embodied chain-of-thought for generalizable robot manipulation. *arXiv preprint arXiv:2606.03784*, 2026b.

Gemini Robotics Team, Saminda Abeyruwan, Joshua Ainslie, Jean-Baptiste Alayrac, Montserrat Gonzalez Arenas, Travis Armstrong, Ashwin Balakrishna, Robert Baruch, Maria Bauza, Michiel Blokzijl, et al. Gemini robotics: Bringing ai into the physical world. *arXiv preprint arXiv:2503.20020*, 2025.

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.

Homer Walke, Kevin Black, Abraham Lee, Moo Jin Kim, Max Du, Chongyi Zheng, Tony Zhao, Philippe Hansen-Estruch, Quan Vuong, Andre He, Vivek Myers, Kuan Fang, Chelsea Finn, and Sergey Levine. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning (CoRL)*, 2023.

Qiuyue Wang, Mingsheng Li, Jian Guan, Jinhui Ye, Sicheng Xie, Yitao Liu, Junhao Chen, Zhixuan Liang, Jie Zhang, Xintong Hu, et al. Qwen-vla: Unifying vision-language-action modeling across tasks, environments, and robot embodiments. *arXiv preprint arXiv:2605.30280*, 2026.

Shihan Wu, Xuecheng Liu, Shaoxuan Xie, Pengwei Wang, Xinghang Li, Bowen Yang, Zhe Li, Kai Zhu, Hongyu Wu, Yiheng Liu, et al. Robocoin: An open-sourced bimanual robotic data collection for integrated manipulation. *arXiv preprint arXiv:2511.17441*, 2025.

Ruihan Yang, Qinxi Yu, Yecheng Wu, Rui Yan, Borui Li, An-Chieh Cheng, Xueyan Zou, Yunhao Fang, Xuxin Cheng, Ri-Zhao Qiu, et al. Egovla: Learning vision-language-action models from egocentric human videos. *arXiv preprint arXiv:2507.12440*, 2025.

Seonghyeon Ye, Joel Jang, Byeongguk Jeon, Se June Joo, Jianwei Yang, Baolin Peng, Ajay Mandlekar, Reuben Tan, Yu-Wei Chao, Bill Yuchen Lin, et al. Latent action pretraining from videos. In *International Conference on Learning Representations*, volume 2025, pp. 28213–28239, 2025.

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.

Wentao Yuan, Jiafei Duan, Valts Blukis, Wilbert Pumacay, Ranjay Krishna, Adithyavairavan Murali, Arsalan Mousavian, and Dieter Fox. Robopoint: A vision-language model for spatial affordance prediction in robotics. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard (eds.), *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 4005–4020. PMLR, 06–09 Nov 2025. URL `https://proceedings.mlr.press/v270/yuan25c.html`.

Michał Zawalski, William Chen, Karl Pertsch, Oier Mees, Chelsea Finn, and Sergey Levine. Robotic control via embodied chain-of-thought reasoning. *arXiv preprint arXiv:2407.08693*, 2024.

Wenyao Zhang, Hongsi Liu, Zekun Qi, Yunnan Wang, Xinqiang Yu, Jiazhao Zhang, Runpei Dong, Jiawei He, He Wang, Zhizheng Zhang, et al. Dreamvla: a vision-language-action model dreamed with comprehensive world knowledge. *Advances in Neural Information Processing Systems*, 38: 24195–24228, 2026.

Qingqing Zhao, Yao Lu, Moo Jin Kim, Zipeng Fu, Zhuoyang Zhang, Yecheng Wu, Zhaoshuo Li, Qianli Ma, Song Han, Chelsea Finn, et al. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1702–1713. IEEE, 2025.

Ruijie Zheng, Dantong Niu, Yuqi Xie, Jing Wang, Mengda Xu, Yunfan Jiang, Fernando Castañeda, Fengyuan Hu, You Liang Tan, Letian Fu, et al. Egoscale: Scaling dexterous manipulation with diverse egocentric human data. *arXiv preprint arXiv:2602.16710*, 2026.

Linqing Zhong, Yi Liu, Yifei Wei, Ziyu Xiong, Maoqing Yao, Si Liu, and Guanghui Ren. Acot-vla: Action chain-of-thought for vision-language-action models. *arXiv preprint arXiv:2601.11404*, 2026.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. Dino-wm: World models on pre-trained visual features enable zero-shot planning. *arXiv preprint arXiv:2411.04983*, 2024.

Xueyang Zhou, Yangming Xu, Guiyao Tie, Yongchao Chen, Guowen Zhang, Duanfeng Chu, Pan Zhou, and Lichao Sun. Libero-pro: Towards robust and fair evaluation of vision-language-action models beyond memorization. *arXiv preprint arXiv:2510.03827*, 2025.

APPENDIX OVERVIEW

- **Section A. Extended Implementation Details**
  - Section A.1. Model Configuration
  - Section A.2. Input and Output Configuration
  - Section A.3. Training Data
  - Section A.4. Training Configuration
  - Section A.5. Details on Future Prediction Loss Design
  - Section A.6. Details on RL Training Objectives
- **Section B. Evaluation Details**
  - Section B.1. Simulation Benchmarks and Baselines
  - Section B.2. Real-Robot Setup
  - Section B.3. Real-Robot Evaluation
- **Section C. Additional Analyses and Results**
  - Section C.1. Additional Qualitative Results on PCA and Attention Analysis
  - Section C.2. Closed-Loop Future-Prediction Results
  - Section C.3. LIBERO-Plus Evaluation Results
  - Section C.4. Inference Efficiency
- **Section D. Limitations and Reproducibility**

A EXTENDED IMPLEMENTATION DETAILS

A.1 MODEL CONFIGURATION

ThinkWorld uses Qwen3-VL-2B-Instruct (Bai et al., 2025) as its vision-language backbone and a frozen DINOv3 ViT-H/16+ target encoder (Siméoni et al., 2025). The DINO encoder receives the main-camera target image at 224×224 resolution and produces a 14×14 grid of 1,280-dimensional patch features. We append 64 learned future-query tokens to the VLM sequence. A two-layer MLP and a six-layer cross-attention decoder map their 2,048-dimensional hidden states to 196 predicted DINO patch features. The future target is the main-camera observation closest to exactly one second after the current observation; samples more than 50 ms from the requested time are masked from the future loss.

Continuous actions are produced by an action expert initialized from MolmoAct2 (Fang et al., 2026). The expert contains 36 non-causal Transformer blocks with hidden width 768 and operates in FP32. It exposes a maximum 30×32 action tensor, but a two-dimensional validity mask restricts each example to its dataset-native horizon and action channels. The expert receives layer-wise Qwen key/value features from input tokens only; generated CoD tokens and future-query tokens are not part of its direct context.

For clean normalized action a and Gaussian noise ϵ, we sample $z \sim \mathrm{Beta}(1, 1.5)$, set $t = 0.001 + 0.999z$, and construct $x_t = (1 - t)\epsilon + ta$. The action expert regresses $a - \epsilon$ using masked mean-squared error over valid entries. We draw eight independently noised flow samples per training example and use ten Euler integration steps at inference.

A.2 INPUT AND OUTPUT CONFIGURATION

Input prompt. All media precede the textual metadata in a single user turn. When temporal context is available, the order is main-view history, current main view, and the remaining current views in their dataset-defined order. History contains at most 16 main-view frames sampled at 2 Hz and ordered from oldest to newest. The current main image is not repeated in the history.

The complete VLA input/output template is shown below. Bracketed lines are optional, and the textual portion is rendered as one normalized string rather than as separate lines.

Future-prediction prompt template

```
USER MEDIA:
[main-camera history video: 0--16 frames]
<current_main_image>
[<current_wrist_image_1><current_wrist_image_2>]

USER TEXT:
The task is to {task}. The setup is {embodiment}
[ with {native_control} control].
[The current state of the robot is {state}.]
[A {N}-frame history video is followed by the current observation image.
| The current observation frame is provided.]
The future visual target is the main-view observation exactly one second
after the current observation. Given these, produce a future motion plan
and then predict the future state one second in the future.

ASSISTANT:
```

Proprioceptive state encoding. Following MolmoAct2 (Fang et al., 2026), we encode proprioceptive state as discrete tokens. Each dataset-native state dimension is normalized using training-split statistics, clipped to $[-1, 1]$, uniformly quantized into 256 bins, and serialized between dedicated state-boundary tokens. The native state width is preserved, so no common padded state vector is exposed in the prompt.

Media processing. Dataset-side resizing is disabled. Qwen retains the source aspect ratio, with a pixel-area range of 4,096–200,704 for each current image and 4,096–50,176 for each history frame. State conditioning is dropped with probability 0.10 and wrist views with probability 0.20 during supervised training. The maximum multimodal sequence length is 2,048 tokens; over-length examples are removed rather than silently truncating their supervision.

Training-time image augmentation can be summarized with the following torchvision-style transform:

Image augmentation pipeline

```
import torchvision.transforms as T

transform = T.Compose([
    T.RandomResizedCrop((height, width), scale=(0.95, 0.95)),
    T.RandomRotation(degrees=5),
    T.ColorJitter(brightness=0.2, contrast=0.2,
                  saturation=0.2, hue=0.05),
    T.RandomApply([
        T.GaussianBlur(kernel_size=5, sigma=(0.1, 1.0))
    ], p=0.2),
])
```

The transform parameters are sampled once and replayed across all views and time steps in an example. RL disables image augmentation so that the four sampled completions for a prompt receive identical media.

CoD tokens receive language cross-entropy supervision, whereas future-query tokens are excluded from cross-entropy and supervise only the future projector. Action SFT uses an empty `<think>...</think>` wrapper and disables assistant-token cross-entropy.

Motion Tokenizer Training. We train a motion tokenizer following the FAST architecture (Pertsch et al., 2025) on one-second windows of consecutive wrist-motion deltas from robot and human trajectories. Each time step contains a 6D motion vector for one hand or a 12D vector for two hands: translation in meters and rotation as a rotation vector in radians, with gripper state excluded. We normalize translation and rotation using fixed scales of 0.3448 m and 0.9119 rad, respectively, and encode the sequences with a vocabulary of 2,048 tokens. Robot poses are expressed in the

Table 4: Training datasets and native observation/action contracts. K and M denote thousands and millions, respectively.

Source	Episodes	Frames	Hours	Hz	Camera view	Action representation
YAM	32.2K	76.0M	704.1	30	Top + two wrists	30×14D absolute joint
DROID	74.6K	17.8M	328.9	15	Exterior + wrist	15×8D absolute joint
RoboCOIN	112.0K	82.3M	755.2	30/50	One or three views	$30/50 \times (12/14)$D absolute EEF
Bridge	53.2K	1.9M	105.2	5	Main	5×7D delta EEF
RT-1	87.2K	3.8M	350.6	3	Main	3×7D delta EEF
BC-Z	39.4K	5.5M	152.0	10	Main	10×7D delta EEF
EgoVerse	55.6K	124.8M	1,155.4	30	Egocentric	One-second 6D/12D motion tokens
EgoDex	314.8K	78.3M	724.8	30	Egocentric	One-second 6D/12D motion tokens
Total	**769.1K**	**390.3M**	**4,276.1**	–	Mixed	Dataset-native

Table 5: Training configuration. Batch is the effective global batch, and allocations report representative resources used for the runs.

Setting	VLM SFT	Action SFT	RL
Initialization	Qwen3-VL-2B	VLM SFT	Action SFT
Trainable modules	Language model + future projector	Action expert	Language model
Objective	CoD/VQA CE $+3\mathcal{L}_{\text{future}}$	Masked flow matching	GRPO/GDPO + weak SFT replay
Data	Robot + human motion; auxiliary VQA	Robot trajectories	Robot + human motion
Iterations	150k	50k (5k current-view + 45k history)	2k
Learning rate	VLM 10^{-5}; projector 10^{-4}	Expert $10^{-4} \rightarrow 5 \times 10^{-5}$	10^{-6}
Schedule	5% warmup, constant	Warmup + cosine cooldown	50-step warmup, constant
Batch	128	128	128
Seed	42	43 / 44	45
Allocation	16×8 GPUs	8×8 then 16×8 GPUs	8×8 GPUs

robot base frame, whereas human poses are expressed in an actor-centered frame whose $+z$ axis is gravity-aligned and whose $+x$ axis follows the actor's current yaw. Training on both domains yields a shared token space for intended wrist motion. The tokens are not executed as native robot actions; a separate continuous action expert predicts executable actions.

A.3 Training Data

VLM SFT combines robot trajectories from BimanualYAM (Fang et al., 2026), DROID (Khazatsky et al., 2024), RoboCOIN (Wu et al., 2025), Bridge (Walke et al., 2023), RT-1 (O'Neill et al., 2024), and BC-Z (Jang et al., 2021) with egocentric human-motion trajectories from EgoVerse (Punamiya et al., 2026) and EgoDex Hoque et al. (2026). Table 4 summarizes the frozen source scale and native action contracts without exposing dataset sampling weights. Episode and frame counts describe source data before the episode-disjoint validation holdout, and a frame denotes one trajectory timestep rather than one image per camera. For Action SFT, we exclude the 50-Hz RoboCOIN trajectories and retain only its 30-Hz trajectories.

Auxiliary visual-language supervision covers image and video captioning, caption question answering, spatial reasoning, pointing, counting, point tracking, robotics-oriented pointing, and robot VQA. Specifically, Pixmo-Cap (Deitke et al., 2025), Molmo2-ER (Fang et al., 2026), Molmo2 Cap-LLaVA, Molmo2 Video-Track (Clark et al., 2026), RoboPoint (Yuan et al., 2025), and RoboVQA (Sermanet et al., 2023). These examples retain their language targets and do not contribute continuous-action or future-feature losses.

A.4 Training Configuration

Training consists of VLM SFT, Action SFT, and RL. All phases use fused AdamW with $(\beta_1, \beta_2) = (0.9, 0.95)$, $\epsilon = 10^{-8}$, weight decay 0.1, gradient-norm clipping at 1.0, and DeepSpeed ZeRO stage 1 without optimizer offload. Table 5 shows the detailed configuration by phase for direct comparison.

VLM SFT and RL execute the VLM in BF16. During Action SFT, Qwen uses BF16 autocast while the action expert is stored and optimized in FP32. For downstream LIBERO evaluation, we full-finetune

the final model on demonstrations from the Object, Goal, Spatial, and LIBERO-10 suites for 50k iterations using masked flow matching, cosine learning-rate decay.

A.5 Details on Future Prediction Loss Design

Following the main text, let $\mathbf{z}_{t+\Delta}$ be the predicted DINO feature map and $\mathbf{z}^\star_{t+\Delta} = \mathcal{E}_{\text{future}}(o_{t+\Delta})$ the target from the observed future, with $\Delta = 1$ s. The distance used in $\mathcal{L}_{\text{future}} = \text{dist}(\mathbf{z}_{t+\Delta}, \mathbf{z}^\star_{t+\Delta})$ is

$$\text{dist}(\mathbf{z}, \mathbf{z}^\star) = \frac{1}{|\mathcal{V}|} \sum_{p \in \mathcal{V}} \left[1 - \cos(\mathbf{z}_p, \mathbf{z}^\star_p) + \lambda_{\text{SL1}} \, \text{SmoothL1}_\beta(\mathbf{z}_p, \mathbf{z}^\star_p)\right], \tag{5}$$

where $\mathcal{V}$ contains the valid image patches. Both feature maps are ℓ_2-normalized before computing the loss. We use $\beta = 0.01$ and $\lambda_{\text{SL1}} = 0.25 \times 21.792413$. During RL, the future projector is frozen and $\mathcal{L}_{\text{future}}$ is not optimized.

A.6 Details on RL Training Objectives

For each context, we sample $K = 4$ CoDs and use the frozen future projector to obtain a predicted future representation from each candidate. The similarity function in the main-text reward $r_{\text{world}} = \text{sim}(\mathbf{z}_{t+\Delta}, \mathbf{z}^\star_{t+\Delta})$ averages global and motion-weighted patch agreement:

$$\text{sim}(\mathbf{z}, \mathbf{z}^\star) = \frac{1}{2} \left[\frac{1}{|\mathcal{V}|} \sum_{p \in \mathcal{V}} s_p + \frac{\sum_{p \in \mathcal{V}} w_p s_p}{\sum_{p \in \mathcal{V}} w_p} \right], \tag{6}$$

where $s_p = \cos(\mathbf{z}_p, \mathbf{z}^\star_p)$. To form w_p, we measure the DINO feature change between the current and observed future frames. The weight is 0.2 for patches at or below the median change, increases linearly to 1 at the 90th percentile, and is clipped outside that range. For nearly static targets, we use the global average for both terms.

A malformed CoD receives a reward 0.05 below the lowest candidate similarity in its group. We standardize rewards within each four-candidate group and then across the rollout batch; each candidate's resulting advantage is shared by its tokens. GRPO uses token-level importance ratios clipped to $[0.8, 1.28]$ and a KL coefficient of 0.04. In the main-text objective $\mathcal{L}_{\text{RL}}$, we set $\lambda_{\text{GRPO}} = 0.999$ and $\lambda_{\text{replay}} = 0.001$. The replay term is teacher-forced cross-entropy on a separate minibatch of canonical CoDs. Only the VLM is updated during RL; the action expert and future projector remain frozen.

B Evaluation Details

B.1 Simulation Benchmarks and Baselines

We fine-tune our policy only on standard LIBERO (Liu et al., 2023) and use its unmodified tasks as the in-distribution reference. We then evaluate the trained policy on LIBERO-PRO and LIBERO-Plus without further adaptation to either benchmark. We select LIBERO-PRO as the primary benchmark because its controlled changes to objects, object positions, task specifications, and environments directly test whether the policy can adapt its behavior to new manipulation settings. LIBERO-Plus additionally covers camera, lighting, background, and sensor-noise perturbations, which probe perceptual robustness beyond our main methodological focus. We report a supplementary LIBERO-Plus evaluation in Appendix C.3.

B.2 Real-Robot Setup

We use the BimanualYAM robot system to collect our in-lab dataset and evaluate trained policies. As shown in Fig. 8, the system comprises two YAM arms and three RealSense D405 cameras: one providing a top view and one mounted on each wrist. Policies use RGB observations only.

Policy inference runs on a single NVIDIA RTX 3090. To account for differences in inference latency, we query each policy and execute its actions at the same 15 Hz control rate, while the remaining system components run at 30 Hz. For each predicted action chunk, we execute the first half and then replan using the latest observation.

Table 6: Full real-robot success-rate (SR) comparison. Each task is evaluated under in-distribution (ID) and out-of-distribution (OOD) settings; cells report successes/trials with SR (%) in parentheses. Task 1 comprises six ID and six OOD conditions (5 trials each); Tasks 2 and 3 comprise three of each. Macro SR is the unweighted mean of the six per-setting rates; Micro SR pools all 120 trials.

	Color Matching		Cup Hanging		Sorting		Overall	
Model	ID	OOD	ID	OOD	ID	OOD	Macro SR	Micro SR
Diffusion Policy	18/30 (60.0)	11/30 (36.7)	9/15 (60.0)	1/15 (6.7)	4/15 (26.7)	1/15 (6.7)	32.8	36.7
MolmoAct2	24/30 (80.0)	8/30 (26.7)	11/15 (73.3)	14/15 (93.3)	12/15 (80.0)	3/15 (20.0)	62.2	60.0
LaWAM	16/30 (53.3)	8/30 (26.7)	9/15 (60.0)	12/15 (80.0)	4/15 (26.7)	4/15 (26.7)	45.6	44.2
ThinkWorld (ours)	**30/30 (100.0)**	**24/30 (80.0)**	**15/15 (100.0)**	**15/15 (100.0)**	**15/15 (100.0)**	**13/15 (86.7)**	**94.4**	**93.3**

B.3 Real-Robot Evaluation

For the real-robot experiments, we evaluate three tasks on the bimanual YAM platform: color matching (60 robot demonstrations), cup hanging (45), and object sorting (100). Within each task, the demonstrations span multiple trajectories and environment configurations, and we use them jointly to fine-tune a single task-specific policy. The tasks are designed to test whether the policy can infer task-level regularities and adapt its actions to the current scene rather than memorize a single trajectory. Separately, we study skill composition on the oven task using 100 robot episodes of atomic-action trajectories and 90 human-video episodes of full task sequences. We co-train a single model on the combined robot and human datasets. Example task trajectory for four tasks can be found in Fig. 9 (a) and Fig. 10 (a).

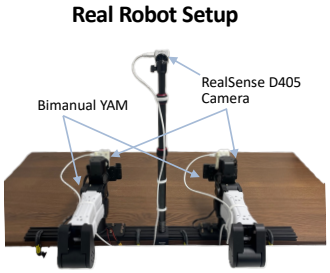


Figure 8: **BimanualYAM setup.**

We fine-tune ThinkWorld for 50,000 steps, applying LoRA to the VLM and fully updating the action DiT decoder. ThinkWorld can also leverage human videos during training; for these samples, we optimize only the VLM future-prediction loss and omit the action DiT loss. We train each baseline for 50,000 steps using its model-specific fine-tuning recipe from the official LeRobot (Cadene et al., 2024) repository.

Tab. 6 reports detailed rollout success rates under in-distribution (ID) and out-of-distribution (OOD) conditions for all tasks. ThinkWorld achieves the highest success rate in all six task–condition pairs, succeeding in all 60 ID trials and 52 of 60 OOD trials. Its macro and micro success rates are 94.4% and 93.3%, respectively, compared with 62.2% and 60.0% for the strongest baseline, MolmoAct2.

C Additional Analyses and Results

C.1 Additional Qualitative Results on PCA and Attention Analysis

We provide additional qualitative visualizations that extend the analysis in the main text. Future-representation PCA and attention visualizations leverage Qwen layers 24–27. PCA and attention are descriptive analyses and are not used as causal evidence.

Fig. 11 expands the PCA visualization of future-prediction representations across five CoD conditions: the original CoD, shuffled CoD, empty CoD, a donor CoD from another episode, and no CoD. This comparison illustrates how the predicted future representations vary with CoD conditioning.

Fig. 12 extends the CoD-to-input-observation attention maps shown in the main text and additionally shows attention from future-prediction tokens to the input observation. CoD attention tends to concentrate on dynamic, task-relevant regions, whereas future-prediction attention is distributed more broadly across the scene. This pattern suggests that CoD emphasizes localized changes relevant to action, while future prediction incorporates wider scene context.

C.2 Closed-Loop Future-Prediction Results

Table 8: **Success rates (%) on LIBERO-Plus across seven perturbation dimensions.** Only methods with published LIBERO-Plus results are listed. Higher is better.

Method	Camera	Robot	Language	Light	Background	Noise	Layout	Avg.
OpenVLA-OFT (Kim et al., 2024)	56.4	31.9	79.5	88.7	93.3	75.8	74.2	69.6
π_0-FAST (Black et al., 2024a)	65.1	21.6	61.0	73.2	73.2	74.4	68.8	61.6
DreamVLA (Zhang et al., 2026)	65.0	40.8	63.5	85.7	82.6	84.9	74.0	69.9
ThinkWorld (Ours)	46.9	70.5	73.5	90.7	90.0	62.7	79.6	**71.7**

Tab. 7 reports SimplerEnv (Li et al., 2024) success rates without additional embodiment-specific fine-tuning for this benchmark. The model nevertheless encountered Bridge and RT-1 data during SFT and RL, providing prior experience with related robot settings. It achieves 49.31% on Google visual matching, 46.54% on the Google-VA variants (micro average), and 41.67% on WidowX visual matching, with an overall micro success rate of 46.51%. These results show that the policy can complete a substantial fraction of SimplerEnv rollouts without benchmark-specific adaptation.

Table 7: **Out-of-the-box performance on SimplerEnv.**

Group	Task	SR (%)
Google-VM		49.31
Google-VA	Pick Coke can	56.00
	Drawer	33.33
	Move-near	55.83
	Put-in-drawer	2.12
	Micro avg.	**46.54**
WidowX VM		41.67
Overall	**micro**	**46.51**

Fig. 13 visualizes future predictions during closed-loop SimplerEnv rollouts. In each column, the one-second-ahead prediction is paired with the observation subsequently reached at the predicted time. The predictions are PCA-projected feature maps and their spatial structure and progression closely track the corresponding observations across the three examples. This qualitative alignment suggests that the future representation remains informative as the policy encounters its own observations and replans under distribution shift.

C.3 LIBERO-Plus Evaluation Results

Tab. 8 reports zero-shot LIBERO-Plus (Fei et al., 2026) results using the policy fine-tuned only on standard LIBERO. ThinkWorld achieves the highest reported Avg. among the four methods at 71.7%, exceeding DreamVLA by 1.8 percentage points. It leads under robot initial-state, lighting, and object-layout perturbations. In particular, ThinkWorld reaches 70.5% on robot initial-state changes, compared with 40.8% for the next-best method, and 79.6% on object-layout changes, compared with 74.2%. These results are consistent with our goal of adapting actions to changed physical configurations and provide supplementary evidence of generalization beyond standard LIBERO.

C.4 Inference Efficiency

Table 9 reports steady-state batch-1 policy latency with CUDA graph capture disabled, using eager inference without static-shape graph setup. All models are measured on the same NVIDIA H100 80GB GPU, with five warm-up calls followed by 50 CUDA-synchronized timed calls. Each model uses its native deployed input contract, so the table reflects practical policy latency rather than an identical-input architecture ablation.

ThinkWorld uses the reasoning process only during training, where it encourages the VLM to learn more action-centric representations.

Table 9: **Inference efficiency.** Per-call latency percentiles, total wall time, and peak GPU memory, measured at steady state over 50 consecutive eager-mode calls on a single NVIDIA H100 GPU. Lower is better for all metrics.

Policy	p50 (ms)	p95 (ms)	Wall (s)	Mem. (GiB)
MolmoAct2	335.07	467.96	18.44	12.11
MolmoAct2-Think	879.37	1014.47	46.76	12.12
ThinkWorld	**324.84**	**333.09**	**16.31**	**7.07**

The deployed policy does not execute this process at inference, so each policy call avoids its additional runtime computation. This separation offers an explanation for how training-time reasoning can support strong task performance while preserving low inference latency.

D LIMITATIONS AND REPRODUCIBILITY

Training results are obtained from a single training seed; accordingly, the reported episode-level confidence intervals characterize evaluation variability rather than variation across independently trained checkpoints. For real-robot experiments, we report the number of physical trials and task-level results to make the evaluation scope explicit. Inference latency should be interpreted under the stated deployment setup, as it varies with hardware and input configuration. To support reproducibility, we provide trial denominators, model initialization details, training configurations, and hardware specifications alongside the aggregate results.

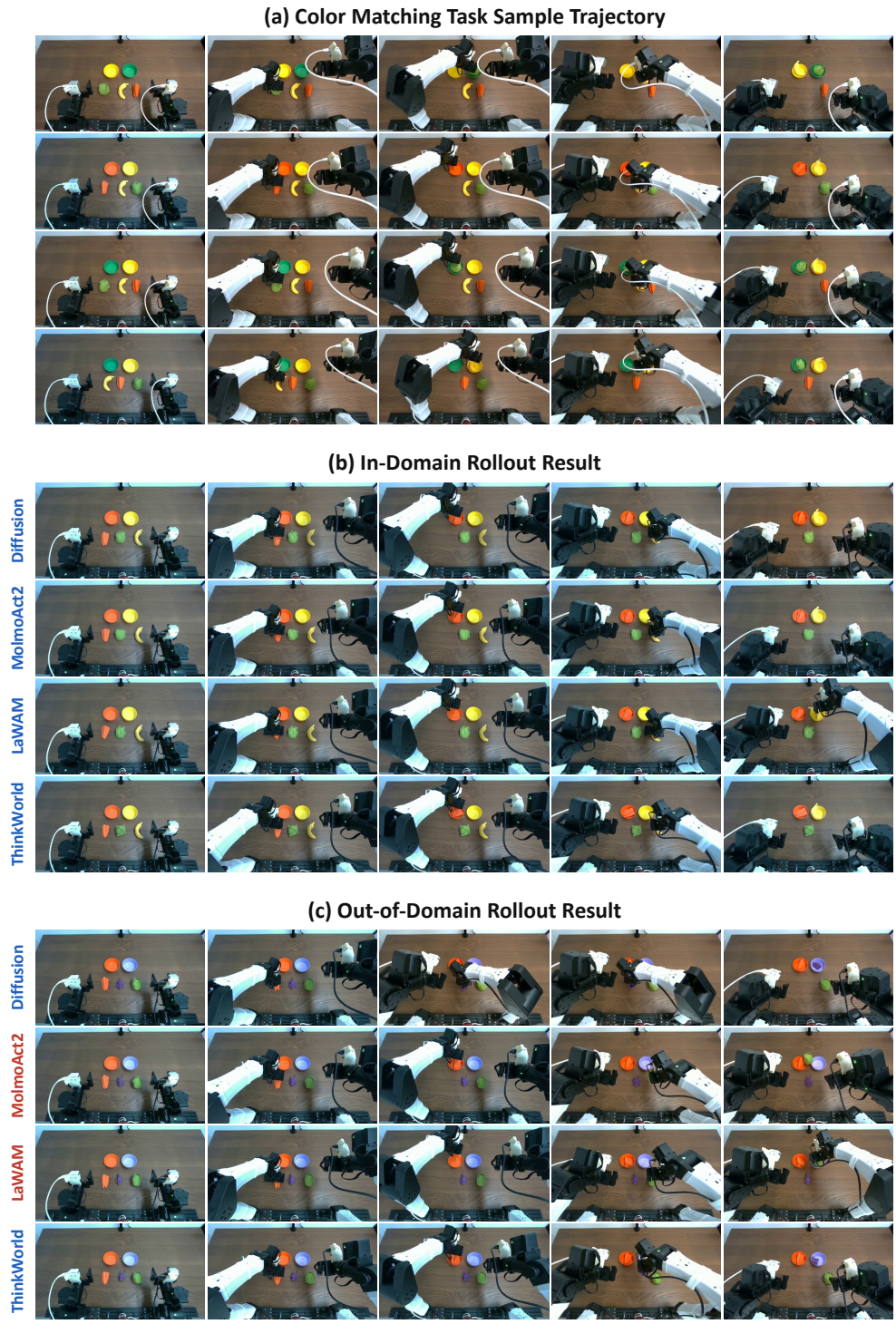


Figure 9: **Real-robot experiment task rollout results.** (a) Sample teleoperation demonstrations used for training. (b) In-domain rollouts. (c) Out-of-domain rollouts. Each panel shows qualitative rollout quality for every method, with frames sampled uniformly over an episode and time advancing left to right; the corresponding success rates are reported in Tab. 6.

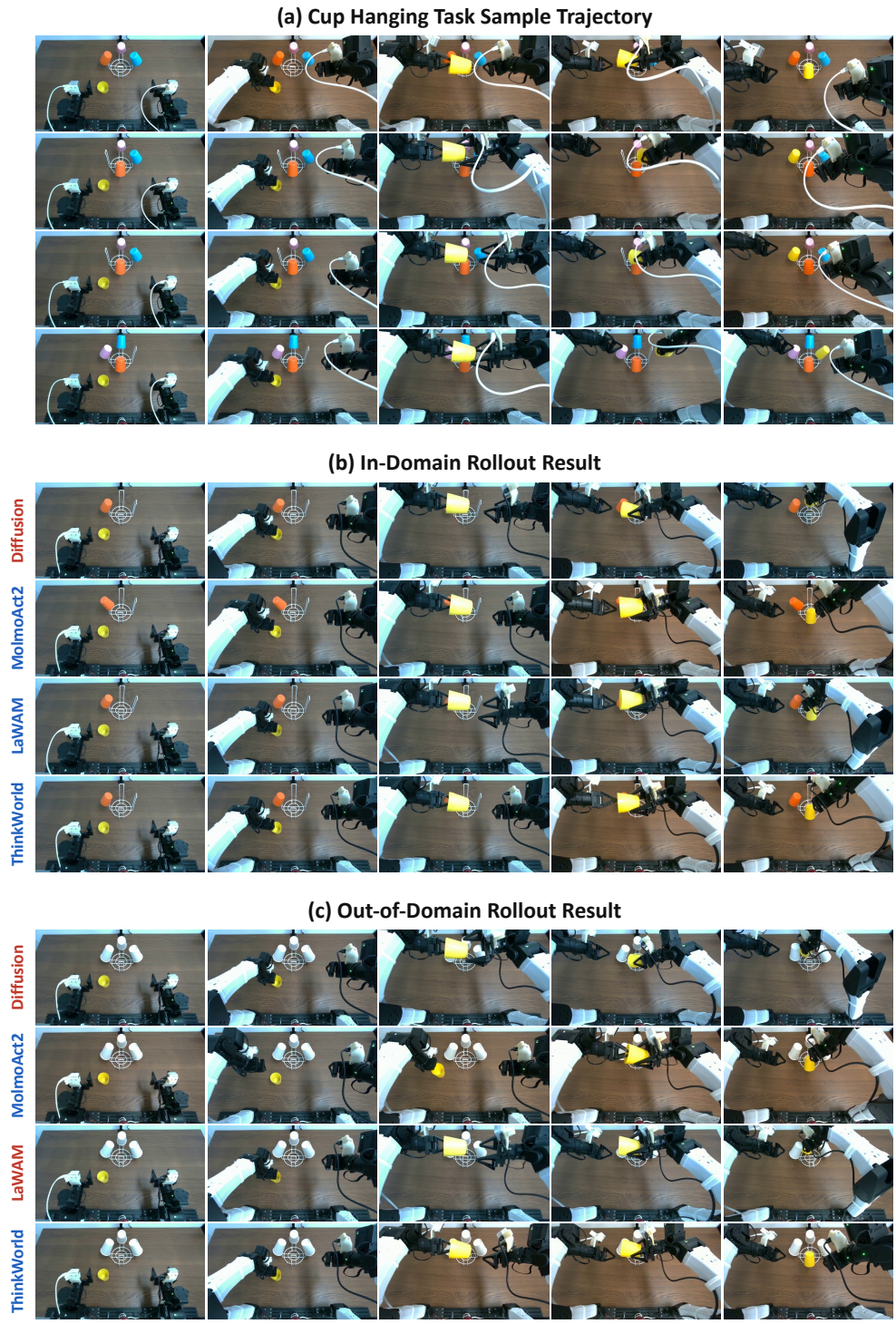


Figure 9: **Real-robot experiment task rollout results.** (a) Sample teleoperation demonstrations used for training. (b) In-domain rollouts. (c) Out-of-domain rollouts. Each panel shows qualitative rollout quality for every method, with frames sampled uniformly over an episode and time advancing left to right; the corresponding success rates are reported in Tab. 6.

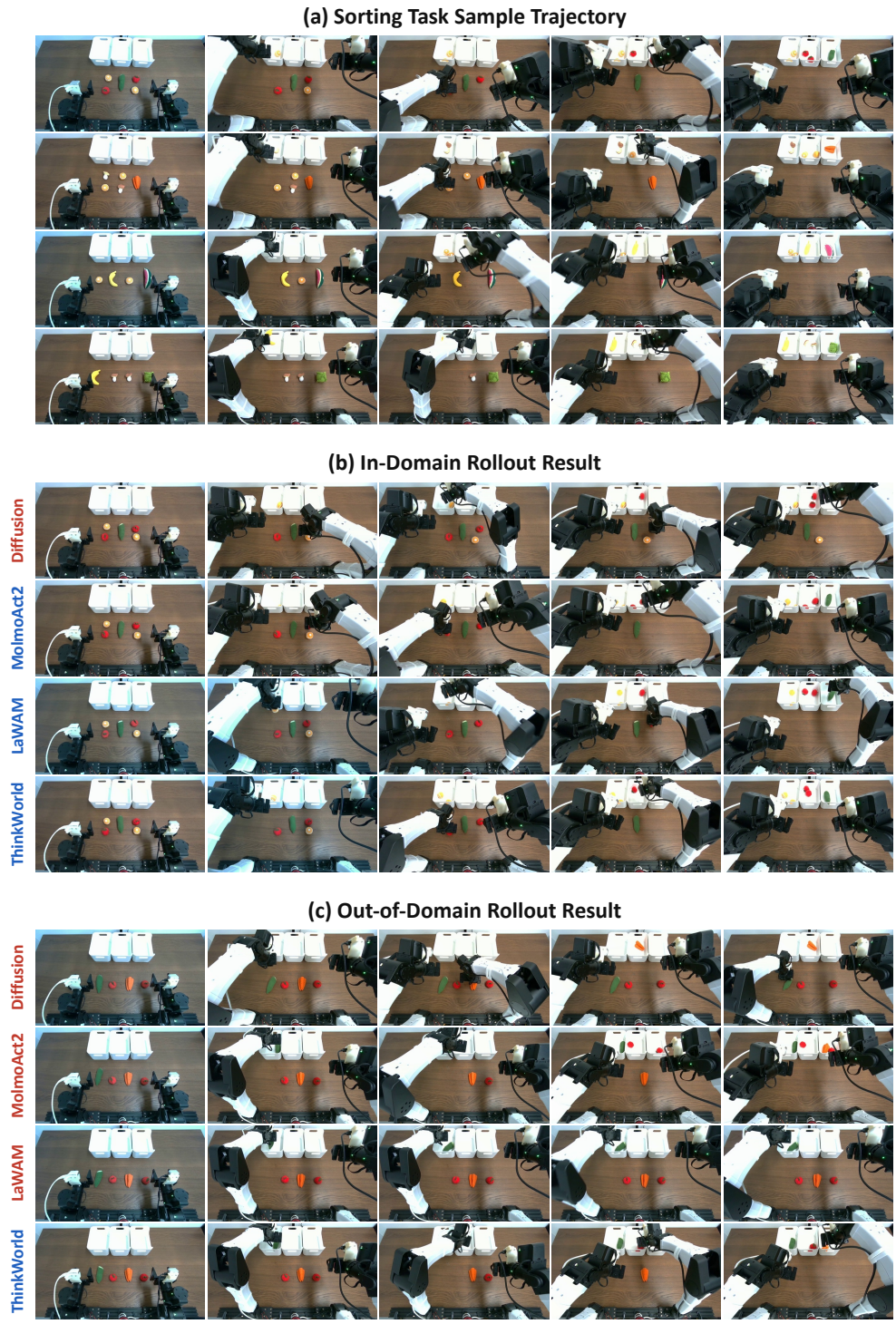


Figure 9: **Real-robot experiment task rollout results.** (a) Sample teleoperation demonstrations used for training. (b) In-domain rollouts. (c) Out-of-domain rollouts. Each panel shows qualitative rollout quality for every method, with frames sampled uniformly over an episode and time advancing left to right; the corresponding success rates are reported in Tab. 6.

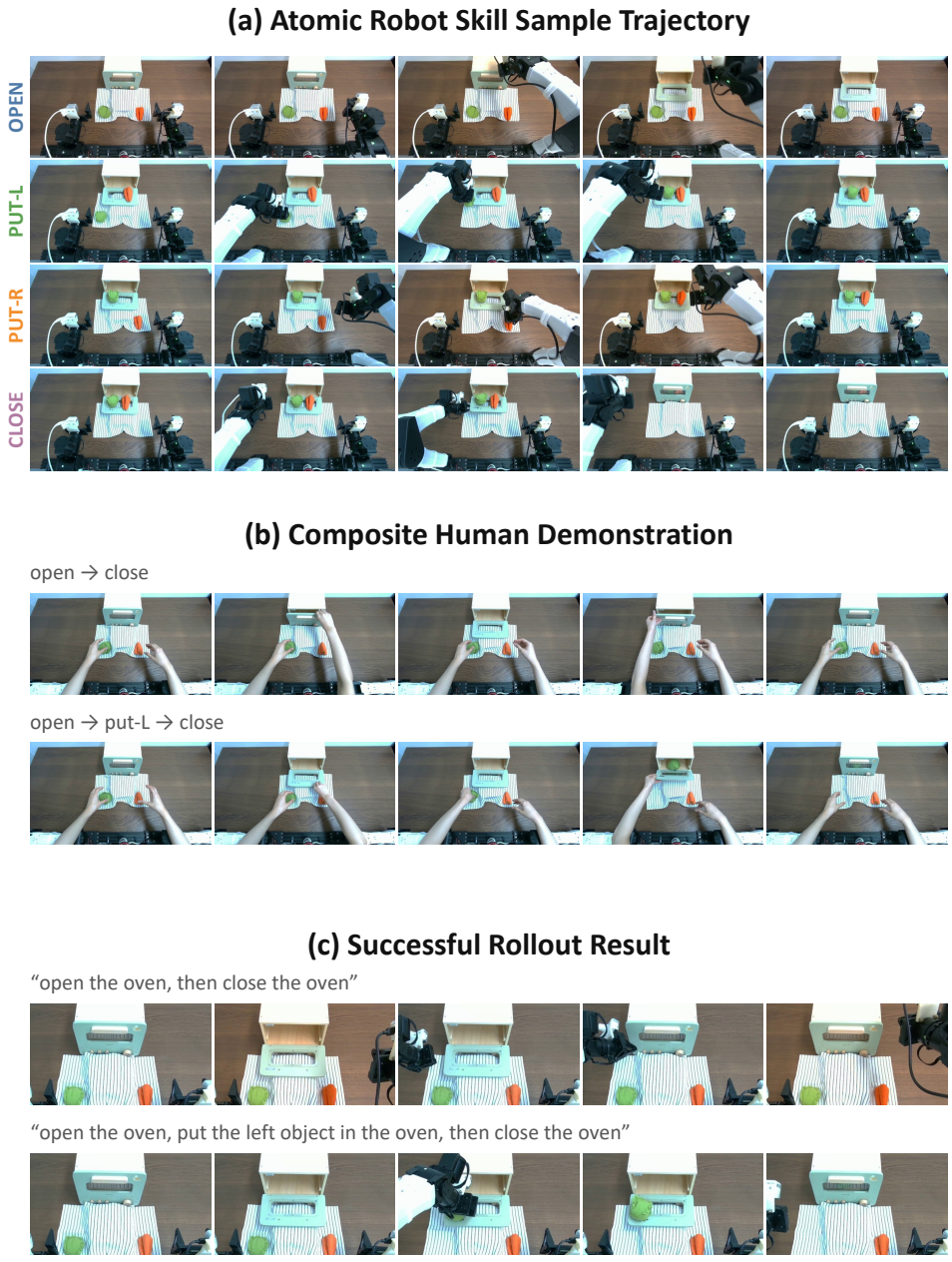


Figure 10: **Real-Robot skill composition sample trajectory and rollout results.** (a) Sample atomic robot trajectories for opening the oven (OPEN), placing the left or right object inside (PUT-L, PUT-R), and closing the oven (CLOSE). (b) Human demonstrations that combine these skills into open $\rightarrow$ close and open $\rightarrow$ put-L $\rightarrow$ close sequences. (c) Successful robot rollouts for the corresponding natural-language instructions. Frames in each row progress from left to right.

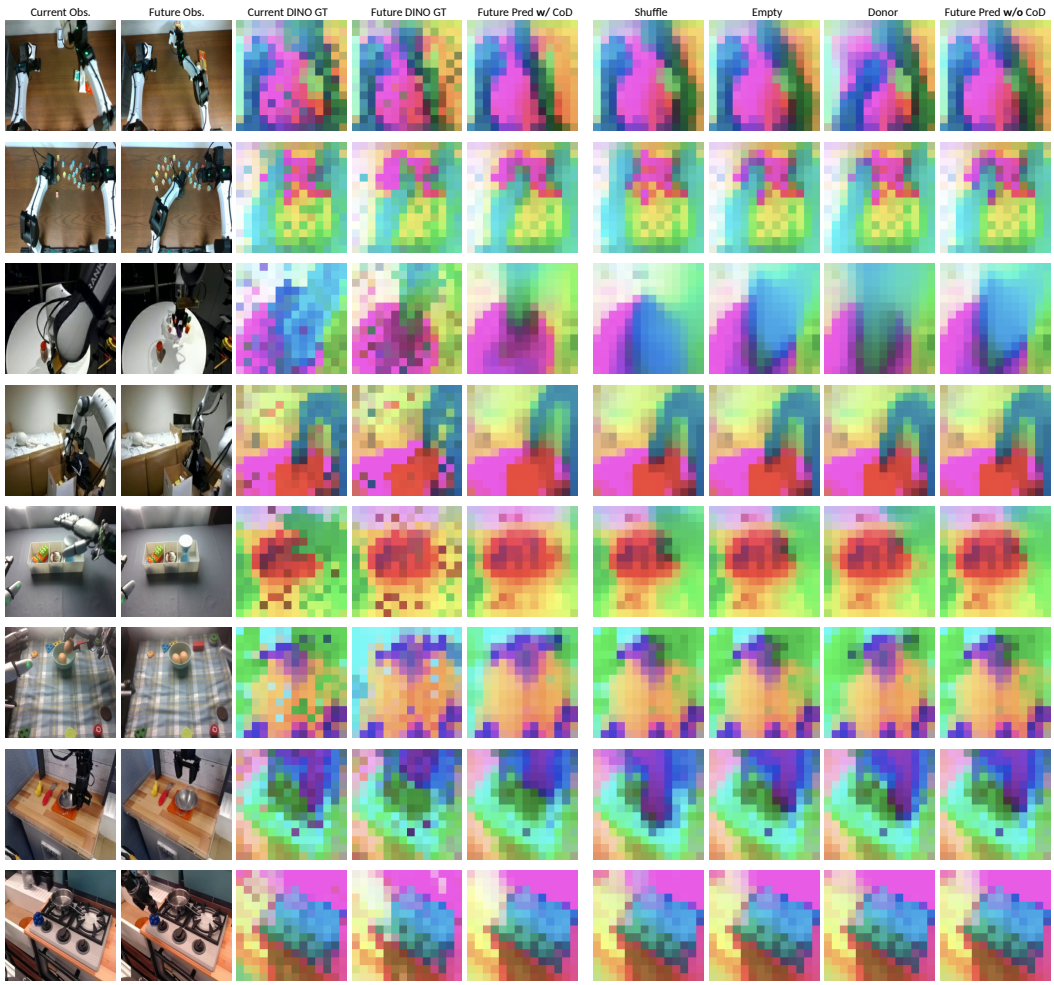


Figure 11: **Qualitative future-prediction results.** Each row is one episode. The first two columns show the current and ground-truth future RGB observations, and the next two show PCA projections of the ground-truth DINO features for those frames. The remaining five columns show the predicted future DINO features under different CoD conditions: with CoD, shuffled CoD, empty CoD, a CoD taken from a different (donor) episode, and without CoD. All PCA panels share one color mapping.

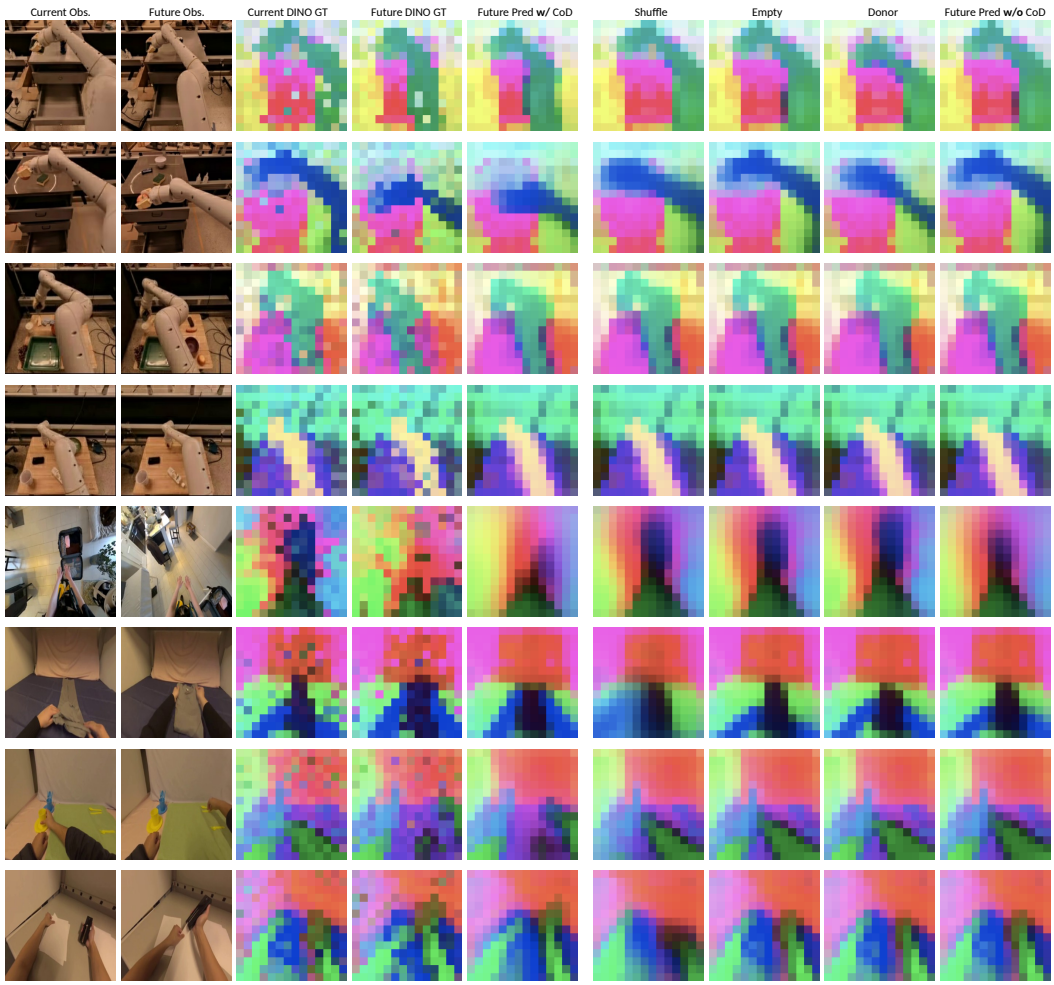


Figure 11: **Qualitative future-prediction results.** Each row is one episode. The first two columns show the current and ground-truth future RGB observations, and the next two show PCA projections of the ground-truth DINO features for those frames. The remaining five columns show the predicted future DINO features under different CoD conditions: with CoD, shuffled CoD, empty CoD, a CoD taken from a different (donor) episode, and without CoD. All PCA panels share one color mapping.

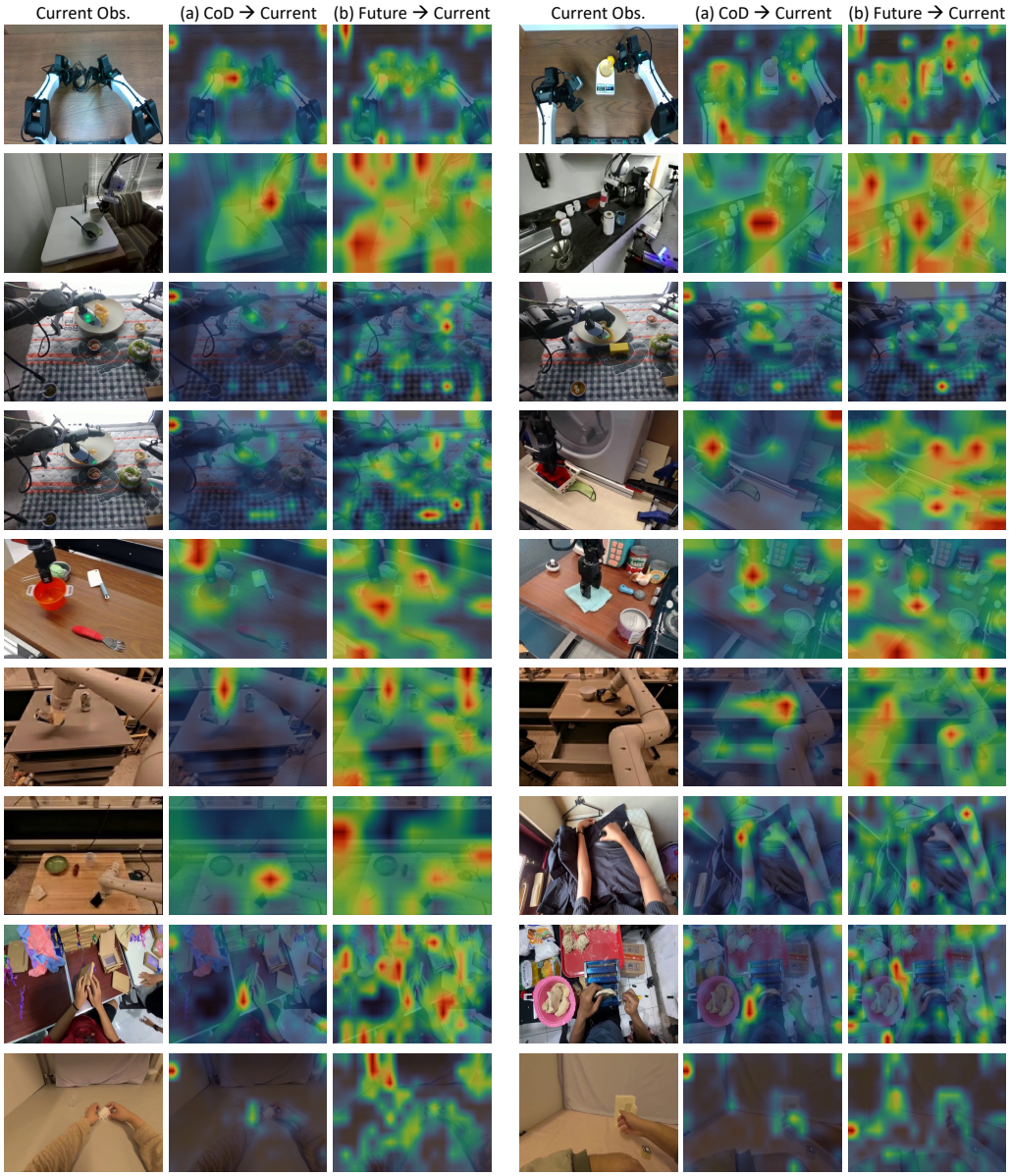


Figure 12: **Additional attention visualizations.** Attention maps from CoD and future-prediction tokens to the input observation. CoD attention concentrates on dynamic, task-relevant regions, while future-prediction attention is more broadly distributed across the scene.

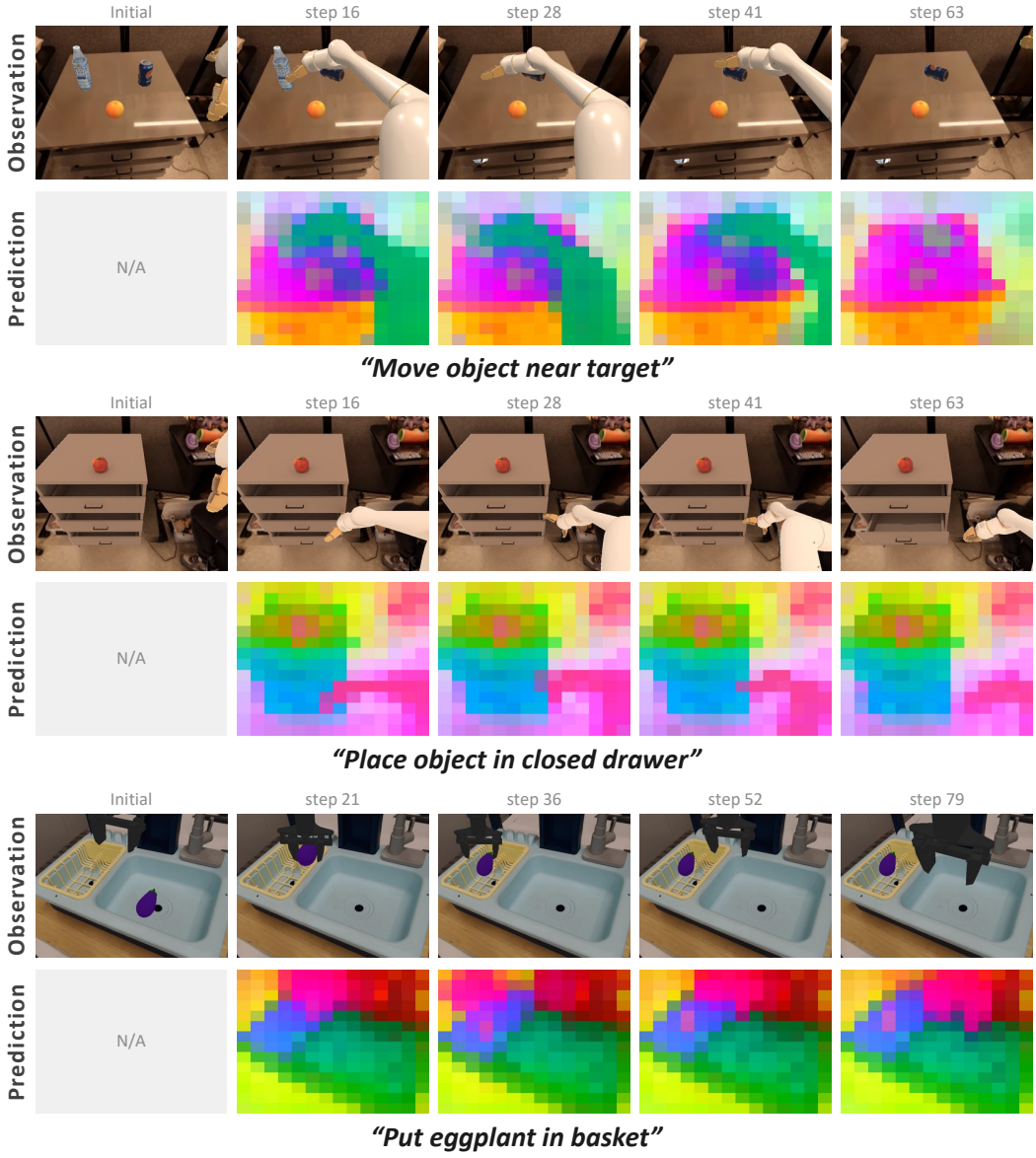


Figure 13: **One-second-ahead future predictions during closed-loop SimplerEnv rollouts.** For each task, the upper row shows observations actually reached by the policy, while the lower row shows PCA-projected future representations predicted at the preceding replanning horizon. Each column pairs a prediction with the observation at its target time, so their scene structure should correspond. The close qualitative alignment in layout and task progress shows that the predictions track the states encountered during closed-loop execution. The initial frame has no preceding prediction.