HUMANOID WORLD ACTION MODEL WITH JOINT STATE–ACTION GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Humanoid robots are a promising platform for general-purpose manipulation. Recent Vision-Language-Action (VLA) policies learn actions directly from multimodal observations, while World Action Models (WAMs) further incorporate future visual prediction to improve action generation. However, in hierarchical humanoid systems, VLA and WAM policies output reference actions that are subsequently realized through whole-body control, robot dynamics, balance, and contact. This hierarchy creates an action–execution gap: the reference produced by the policy can differ from the motion realized by the robot. Without explicitly modeling the realized body state, future visual prediction must jointly explain scene evolution and discrepancies between reference actions and executed motion, making it difficult to associate an action with its physical outcome. We propose HWAM, a Humanoid World Action Model with joint state–action generation, which makes the robot's post-execution proprioceptive state an explicit prediction target. By jointly generating reference actions and their realized body states, HWAM directly incorporates supervision of executed motion into action learning. HWAM is trained through three complementary conditional paths. The Policy path jointly denoises state–action trajectories conditioned only on current observations, matching deployment conditions. Forward Dynamics Modeling (FDM) predicts future visual observations conditioned on actions and post-execution states, while Inverse Dynamics Modeling (IDM) reconstructs the joint trajectory from visual transitions. Together, these paths connect policy references, realized body motion, and visual outcomes. HWAM achieves the highest success rate among evaluated baselines on three real-robot tasks on the LimX OLI humanoid, without pretraining on additional robot data. On Candy Picking, HWAM achieves a 70.6% success rate, compared with 62.5% for $\pi_{0.5}$ and 43.3% for Fast-WAM.

1 INTRODUCTION

General-purpose humanoid robots are expected to understand human instructions and execute diverse tasks in complex physical environments. Recent Vision-language-action (VLA) models have become a promising approach to this problem by unifying visual perception, language understanding, and action generation in a single multimodal policy. Systems such as PaLM-E, RT-2, and OpenVLA show that semantic knowledge from large vision-language models can support robot control and transfer across tasks and scenes (Driess et al., 2023; Zitkovich et al., 2023; Kim et al., 2025). World Action Models (WAMs) further introduce future visual or latent-state prediction into action generation, allowing predicted futures to provide additional supervision or information for control (Shen et al., 2026; Zhu et al., 2025; Ye et al., 2026). These advances have moved robot policies beyond pure action regression toward predictive and generative control.

General-purpose humanoid manipulation requires both task-level motion decisions and execution-level whole-body dynamics control. We adopt a hierarchical control architecture: a high-level vision–language–action policy generates reference trajectories in kinematic space from visual observations, language instructions, and the current proprioceptive state, while a low-level whole-body tracking controller uses real-time proprioceptive feedback to adapt these references to execution dynamics and drives the robot through joint servoing. This division allows the high-level policy to learn task behavior from demonstrations that primarily provide poses and motion trajectories, while reusing the low-level controller's whole-body coordination and feedback control capabilities (Fu et al., 2024;

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.

He et al., 2025). Accordingly, the action a_t in this paper denotes a motion reference supplied to the low-level tracking controller, rather than a torque applied directly to an actuator, and it is not assumed to be realizable point by point. Figure 1 summarizes this hierarchical control interface.

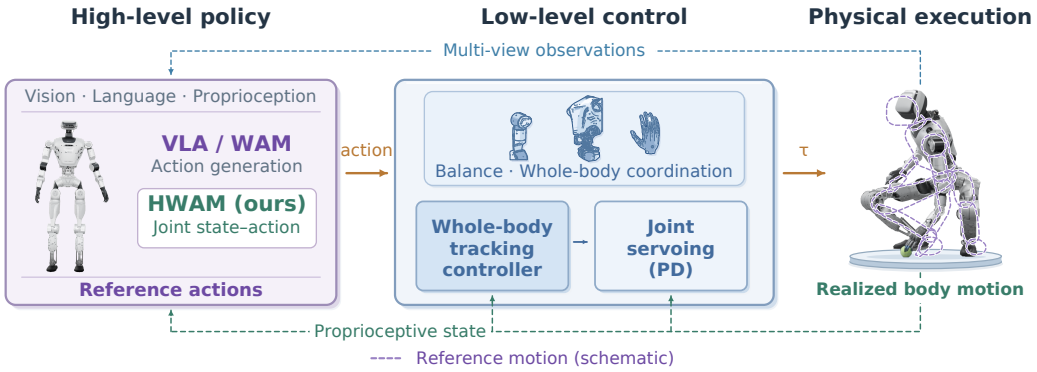

Figure 1: **Hierarchical control interface in HWAM.** The high-level VLA/WAM/HWAM policy produces reference actions, which are adapted by low-level whole-body tracking and joint servoing before physical execution. The measured post-execution proprioceptive state provides feedback to the high-level policy.

This hierarchical interface also introduces a discrepancy between the high-level reference and the actual body response. The low-level tracking policy adapts reference motion to the current body state, contact conditions, and dynamic constraints, so the realized motion can deviate from the reference. For example, a reference trajectory may request simultaneous trunk motion and arm extension; adjustments to trunk motion made by the low-level controller to maintain balance can also change the hand trajectory at contact. Thus, even a kinematically feasible reference does not ensure the intended task outcome after low-level execution. The high-level policy needs to learn the body responses produced by low-level tracking and adaptation to robot dynamics, so that reference generation can account for actual execution outcomes. We refer to the discrepancy between a high-level reference action and its physical realization as the *action–execution gap*.

This gap complicates action learning through future visual prediction. A policy reference can differ from the state realized after low-level execution. If this intermediate state is not modeled, the model must learn simultaneously how the policy action is physically realized and how the realized motion changes the scene. This makes the relation between the policy action and its future visual outcome difficult to learn. HWAM addresses this problem by explicitly placing the post-execution state in the action-generation target and then connecting the joint state–action trajectory to visual transitions in both directions.

Dataset statistics reveal larger discrepancies between commanded joint references and measured proprioceptive states in mobile humanoid manipulation than in stationary bimanual manipulation, including under steady-state conditions (Section 4.1). This observation motivates representing the post-execution proprioceptive state explicitly instead of asking a policy to relate a high-level action directly to its downstream visual consequence.

Prior work has explored body-state representations and prediction for humanoid control (Qiu et al., 2025; Bai et al., 2026). We propose HWAM, a humanoid-oriented World Action Model that incorporates post-execution states into its trajectory generation target. Joint generation is a natural extension of the WAM action stream: state and action share a temporal structure and can be modeled by the same trajectory generator. At each temporal position, we align action a_{t+h} with proprioceptive state s_{t+h+1} and represent them as the joint state–action target

$$\mathbf{Y}_t[h] = [s_{t+h+1}; a_{t+h}]. \tag{1}$$

The state and action channels are concatenated before token projection and jointly denoised by a shared state–action generator. This gives action generation access to the evolving state estimate

throughout denoising. We study how the resulting joint target interacts with visual training objectives to improve closed-loop task performance.

Building on conditional world-action modeling (Zhu et al., 2025; NVIDIA, 2026a), HWAM incorporates post-execution states into three training paths. The Policy path jointly denoises state–action trajectories conditioned on current visual observations, language, and proprioception, without generating future video. This supervises policy references together with their measured execution outcomes. Forward Dynamics Modeling (FDM) supplements this objective by conditioning future visual prediction on actions and post-execution states, explicitly accounting for realized body motion when predicting scene changes. Inverse Dynamics Modeling (IDM) adds supervision in the reverse direction, using visual transitions to jointly predict actions and the body states they produce. The three paths connect policy references to both execution states and visual outcomes. At deployment, predicted states continue to participate in joint denoising, while only the actions are sent to the low-level controller.

Our evaluation asks whether the action–execution gap is observable in robot data, whether a joint state–action target becomes effective when combined with bidirectional visual grounding, and whether the resulting policy improves real-robot manipulation. We evaluate stationary and mobile manipulation on the LimX OLI humanoid and a task-specific clothes-folding setting on ALOHA (Zhao et al., 2023). Our main contributions are:

1. **Post-execution state–action generation.** We pair each high-level action with its temporally aligned post-execution proprioceptive state and denoise the two channels as one joint target.
2. **Execution-state grounding.** Post-execution states serve as conditions for future-video prediction in FDM and as joint prediction targets with actions in IDM and Policy, connecting execution-state supervision to visual transitions.
3. **Controlled real-robot evidence.** A factorial ablation identifies the interaction between the joint target and the three-path training formulation, and real-robot experiments evaluate the resulting model on humanoid and bimanual manipulation.

2 Related Work

2.1 Vision-Language-Action Models

Vision-language-action (VLA) models unify visual perception, language understanding, and robot control. PaLM-E integrates embodied observations into a multimodal language model for reasoning and planning (Driess et al., 2023). RT-2 transfers vision-language knowledge to robot control via action tokenization, while OpenVLA explores open-source VLA learning from large-scale demonstrations (Zitkovich et al., 2023; Kim et al., 2025). Recent systems improve generalization through heterogeneous data and continuous action generation. For instance, $\pi_{0.5}$ achieves open-world generalization via multi-source co-training, enabling manipulation in novel environments (Physical Intelligence et al., 2025), and its successor $\pi_{0.7}$ introduces a steerable generalist foundation model with emergent compositional generalization and cross-embodiment transfer (Physical Intelligence et al., 2026). In NVIDIA's GR00T series, GR00T N1 is an open foundation model for generalist humanoid reasoning and skills, combining a pretrained vision-language module with a diffusion Transformer action generator for cross-embodiment manipulation (NVIDIA et al., 2025). Its updates, GR00T N1.5 and N1.7, enhance instruction following and upgrade the VLM backbone to Cosmos-Reason2-2B (Qwen3-VL) with large-scale human and simulation pretraining, improving generalization and long-horizon reasoning (NVIDIA, 2025; 2026b).

2.2 World Action Models and Joint Generative Policies

World Action Models incorporate future prediction into policy learning. Existing approaches include cascaded models that first generate future observations and then infer actions, as well as joint models that generate future visual variables and actions within a shared generative framework (Shen et al., 2026). Unified World Models (UWM) couple video and action diffusion and use modality-specific diffusion steps to represent policies, forward dynamics, inverse dynamics, and video generation (Zhu et al., 2025). Cosmos 3 distinguishes forward dynamics, inverse dynamics, and a policy mode that

jointly predicts video and action (NVIDIA, 2026a). Motus integrates understanding, video, and action experts in a Mixture-of-Transformers architecture and supports multiple conditional generation modes within one model (Bi et al., 2025). Fast-WAM retains video co-training while predicting actions from current observations without explicit future-video synthesis at inference time (Yuan et al., 2026).

Joint generative trajectory modeling provides a related precedent. Diffuser models state–action trajectories with a diffusion model and generates complete trajectories through iterative denoising (Janner et al., 2022). Cosmos Policy jointly predicts actions and a future state containing proprioception and images (Kim et al., 2026).

2.3 Body-State Representations for Humanoid Control

Body-state representations help humanoid policies coordinate high-dimensional configurations, multiple effectors, and contact-dependent motion. HumanPlus and OmniH2O use hierarchical control to connect task-level motion references with whole-body execution (Fu et al., 2024; He et al., 2025). Qiu et al. (2025) use a shared state–action representation to learn humanoid policies from human and robot demonstrations. HEX combines humanoid-aligned body-part representations with a Unified Proprioceptive Predictor and fuses predictive body-state features with visual-language information for whole-body control (Bai et al., 2026). These works establish body state as a useful representation for transferring motion knowledge and learning coordinated behavior.

3 Method

3.1 Hierarchical Control Interface and Overview

We adopt a hierarchical interface in which a high-level VLA/WAM policy outputs kinematic reference actions a_t (rather than direct actuator torques), and a low-level whole-body tracking controller adapts them using proprioceptive feedback. We write s_{t+1} for the proprioceptive state measured after execution. HWAM treats each reference action and its aligned post-execution state as a joint prediction target, generated by a shared state–action generator. A video expert grounds this joint trajectory in visual transitions through path-dependent mixed attention. Training uses three conditional paths: FDM predicts future video from a demonstrated joint state–action trajectory, IDM predicts the joint trajectory from visual evidence, and Policy generates only the state–action trajectory conditioned on current observations, without generating future video. Figure 2 gives an overview.

3.2 Joint Post-Execution State–Action Chunk Target

Under the hierarchical interface above, the high-level action a_t is a motion reference, while s_{t+1} is the proprioceptive state measured after low-level execution. Task success therefore depends on the state actually reached after execution; an action that is valid in kinematic space is not sufficient.

At time t, the robot receives synchronized multi-view observations $O_t = \{o_t^{(v)}\}_{v=1}^{V}$, a language instruction L, and its measured proprioceptive state s_t. For training targets, t indexes synchronized demonstration samples: a_t denotes the high-level motion reference sent to the low-level tracking controller, and s_{t+1} is the proprioceptive state recorded at the next sample. This naturally couples the action a_t with its execution outcome s_{t+1}.

For a prediction horizon of H steps, we define the action and state chunks as

$$\mathbf{A}_t = [a_t, \ldots, a_{t+H-1}], \qquad \mathbf{S}_t = [s_{t+1}, \ldots, s_{t+H}]. \tag{2}$$

We adopt a simple and effective formulation for joint prediction: the model learns to predict a joint state–action chunk by concatenating the state and action targets along the feature dimension. Let $\mathbf{Y}_t$ be the joint target:

$$\begin{aligned} \mathbf{Y}_t &= \mathrm{Concat}(\mathbf{S}_t, \mathbf{A}_t) \in \mathbb{R}^{H\times(d_s+d_a)}, \\ \mathbf{Y}_t[h] &= [s_{t+h+1}; a_{t+h}], \qquad h = 0, \ldots, H-1. \end{aligned} \tag{3}$$

HWAM places the post-execution state in the same generation target as the action, extending the WAM trajectory stream with state channels. The two are concatenated before token projection and

modeled through a shared trajectory representation. Thus, at every denoising step, the action channels can access the current partially denoised state channels, while state supervision can influence the representation used for action generation.

This construction provides post-execution state supervision through a shared state–action representation. It does not explicitly learn a low-level dynamics model or impose a separate physical-consistency constraint. Moreover, the joint target is not assumed to be useful in isolation: the next subsection describes how it is grounded in visual transitions.

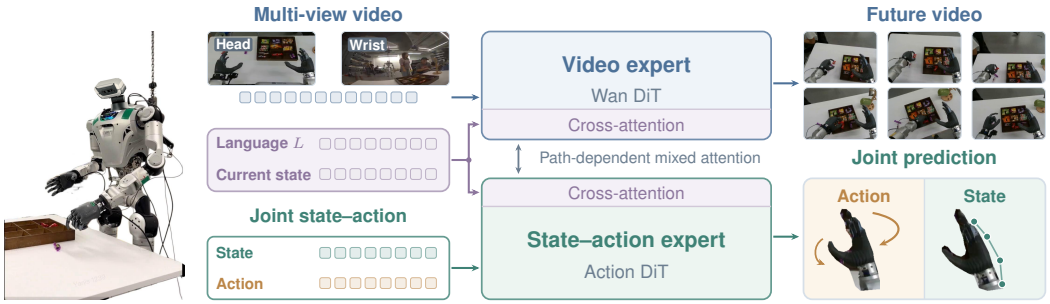

Figure 2: **Overview of HWAM.** The high-level action a_t is executed through the low-level whole-body controller, producing the measured post-execution state s_{t+1}. HWAM represents the joint state–action pair as a single token. A frozen causal video VAE provides current and future visual latents, while the video and state–action experts interact through path-dependent mixed attention.

3.3 Architecture

HWAM consists of two interacting experts: a video expert that models visual transitions and a state–action expert that generates $\mathbf{Y}_t$. Given a synchronized multi-view clip $\mathbf{O}_t^V$, a frozen causal Wan VAE (Team Wan et al., 2025) produces the current and future visual latents $\mathbf{Z}_t^0$ and $\mathbf{Z}_t^+$, denoted collectively by $\mathbf{Z}_t = [\mathbf{Z}_t^0; \mathbf{Z}_t^+]$. The language instruction L and the current proprioceptive state s_t are encoded as the shared conditioning context $\mathbf{C}_t$. Both experts receive $\mathbf{C}_t$, while their access to $\mathbf{Z}_t^0$, $\mathbf{Z}_t^+$, and $\mathbf{Y}_t$ is determined by the selected training path.

The state–action expert extends the action stream used in action-only WAMs. It operates on the target $\mathbf{Y}_t = \mathrm{Concat}(\mathbf{S}_t, \mathbf{A}_t)$ defined above, where each temporal token contains the post-execution state and its corresponding action. The state and action channels are projected into a shared trajectory representation and jointly denoised with the same state–action time τ_{SA}. Consequently, the partially denoised state channels can influence action generation throughout denoising within the shared trajectory stream.

The video and state–action experts use separate Transformer parameters and exchange information through path-dependent masked cross-attention, following the mixed-expert interface of prior WAMs (Yuan et al., 2026). The masks determine the visibility of $\mathbf{Z}_t^0$, $\mathbf{Z}_t^+$, and $\mathbf{Y}_t$ in the FDM, IDM, and Policy paths, allowing the same architecture to support trajectory-to-video, vision-to-trajectory, and deployment-matched policy learning. The detailed visibility patterns and objectives are described in the next subsection.

3.4 Bidirectional State–Action–Vision Grounding

The joint state–action target is trained under three conditional paths that share the same experts and parameters but differ in generation targets and visible conditions. Let $\mathcal{V}_0 = \mathbf{Z}_t^0$, $\mathcal{V}_+ = \mathbf{Z}_t^+$, and $\mathcal{J} = \mathbf{Y}_t$ denote current-video tokens, future-video tokens, and joint state–action tokens. Figure 3 summarizes the three paths and their attention masks.

Training roles. The Policy path conditions only on current observations and generates the joint state–action trajectory without future video, matching deployment. FDM predicts future video from actions and realized post-execution states, so visual prediction accounts for physical execution rather

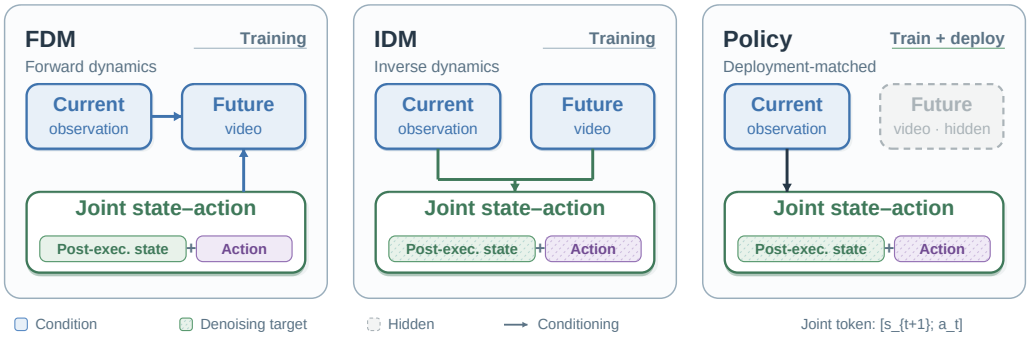

Figure 3: **Path-dependent corruption and attention.** FDM predicts future visual observations from the joint state–action trajectory, whereas IDM reconstructs the joint trajectory from visual transitions, providing complementary bidirectional grounding between visual dynamics and state–action trajectories. Policy matches deployment by conditioning only on current observations.

than the command alone. IDM reverses this direction by predicting actions and post-execution states from visual transitions.

Together, these paths make the post-execution state both an input for explaining visual outcomes and a supervised output coupled to action generation. The factorial ablation in Section 4.5 evaluates this combined training design against VGM/Policy, which combines video generation modeling (VGM) with Policy training.

Conditional flow-matching objectives. We use conditional flow matching (Lipman et al., 2023) for both generated modalities. For a clean target $\mathbf{X}$, Gaussian noise $\boldsymbol{\epsilon}_X \sim \mathcal{N}(0, \mathbf{I})$, and noise level $\tau \in [0, 1]$, the corrupted target and target velocity are

$$\mathbf{X}_\tau = (1-\tau)\mathbf{X} + \tau\boldsymbol{\epsilon}_X, \qquad \mathbf{U}_X = \boldsymbol{\epsilon}_X - \mathbf{X}. \tag{4}$$

We apply this construction to the future-video target $\mathbf{Z}_t^+$ and the joint trajectory target $\mathbf{Y}_t$, with independent noise levels τ_V and τ_{SA} and corresponding velocities $\mathbf{U}_V$ and $\mathbf{U}_{SA}$. State and action channels share τ_{SA}. We use elementwise mean squared error, denoted by MSE, for each generated target. State and action errors are averaged together without additional channel or timestep weights.

The clean contextual conditions for each path are $\mathbf{Q}_F = (\mathbf{Z}_t^0, \mathbf{Y}_t, \mathbf{C}_t)$, $\mathbf{Q}_I = (\mathbf{Z}_t^0, \mathbf{Z}_t^+, \mathbf{C}_t)$, and $\mathbf{Q}_P = (\mathbf{Z}_t^0, \mathbf{C}_t)$. The video and state–action velocity predictors, denoted by v_θ^V and v_θ^{SA}, process the noisy target conditioned on both its noise level and the path-specific context $\mathbf{Q}_m$. Their losses are

$$\begin{aligned}
\mathcal{L}_{\mathrm{F}} &= \mathbb{E}\big[\mathrm{MSE}\big(v_\theta^V(\mathbf{Z}_{t,\tau_V}^+ \mid \tau_V, \mathbf{Q}_F), \mathbf{U}_V\big)\big], \\
\mathcal{L}_{\mathrm{I}} &= \mathbb{E}\big[\mathrm{MSE}\big(v_\theta^{SA}(\mathbf{Y}_{t,\tau_{SA}} \mid \tau_{SA}, \mathbf{Q}_I), \mathbf{U}_{SA}\big)\big], \\
\mathcal{L}_{\mathrm{P}} &= \mathbb{E}\big[\mathrm{MSE}\big(v_\theta^{SA}(\mathbf{Y}_{t,\tau_{SA}} \mid \tau_{SA}, \mathbf{Q}_P), \mathbf{U}_{SA}\big)\big].
\end{aligned} \tag{5}$$

The expectations are over demonstration samples, Gaussian noise, and the noise levels sampled for the generated modality. FDM corrupts and supervises only future-video targets; IDM and Policy corrupt and supervise the joint trajectory. Conditioning variables remain clean and are excluded from the loss. In Policy, future-video tokens are hidden and only state–action tokens are denoised, during both training and deployment.

We sample one path $m \in \{\mathrm{F}, \mathrm{I}, \mathrm{P}\}$ per minibatch, with $\pi_m = 1/3$ and $\lambda_m = 1$, and optimize

$$\mathcal{L}_{\mathrm{HWAM}} = \mathbb{E}_{m\sim\pi}\left[\lambda_m \mathcal{L}_m\right] \quad = \tfrac{1}{3}(\mathcal{L}_{\mathrm{F}} + \mathcal{L}_{\mathrm{I}} + \mathcal{L}_{\mathrm{P}}). \tag{6}$$

4 Experiments

We evaluate HWAM on three real-robot tasks with the LimX OLI humanoid and on offline open-loop diagnostics. We compare with vision–language–action (VLA) and world-model-based policies, and conduct a controlled ablation on the mobile manipulation task to examine the joint state–action target with FDM/IDM/Policy training. ALOHA results are reported in Appendix D.

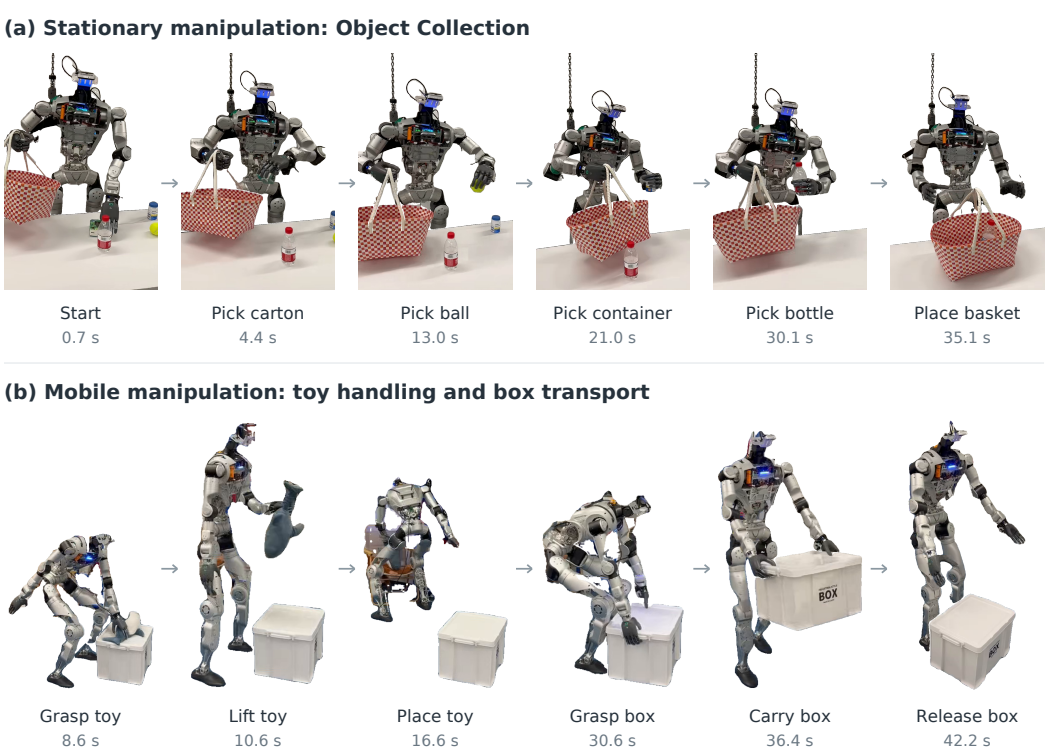

Figure 4: **Examples from the OLI evaluation suite.** The stationary manipulation example shows repeated tabletop object collection, while the mobile manipulation example combines object handling with changes in body posture and position. Each row follows a separate video timeline; backgrounds are simplified and panels are cropped for visualization.

4.1 Action–Execution Gap in Robot Data

Before evaluating policy performance, we quantify how high-level actions are realized in recorded robot trajectories. We compare each commanded joint reference a_t with its aligned post-execution proprioceptive state s_{t+1} on the LimX OLI mobile humanoid and the stationary ALOHA bimanual platform. Both settings show non-zero action–state errors, including residual tracking errors under approximately constant joint references (LimX OLI: 4.420° direct, 2.885° steady-state; ALOHA: 0.648° direct, 0.402° steady-state). These data motivate modeling the post-execution state rather than assuming exact tracking. The full measurement protocol, alignment rules, and per-joint statistics are provided in Appendix A.

4.2 Experimental Setup

Platform, tasks, and data. We use the LimX OLI, a full-size humanoid with a 31-DoF whole-body action space (including two 7-DoF arms) and BrainCo dexterous hands. The OLI suite contains two stationary manipulation tasks, *Candy Picking* and *Object Collection*, and one mobile whole-body manipulation task, *Plush-Toy Picking*. placement, repeated reaching, and mobile whole-body coordination. The task-specific training sets contain 1,783, 4,220, and 404 episodes, respectively. Candy Picking has an additional held-out validation set of 422 episodes for offline diagnostics; these episodes are excluded from training. Further task definitions, control demands, and completion criteria are given in Appendix B.

Baselines. We compare HWAM with five policies on OLI: GR00T N1.5 (NVIDIA, 2025), $\pi_{0.5}$ (Physical Intelligence et al., 2025), Cosmos3-Edge (NVIDIA, 2026a), DiT4DiT (Ma et al., 2026), and Fast-WAM (Yuan et al., 2026). Fast-WAM is the closest prior WAM baseline because it also uses video co-training while generating actions without future-video synthesis at deployment. The ALOHA appendix additionally reports results for π_0 (Black et al., 2024) and SmolVLA (Shukor et al., 2025).

Evaluation protocol. On OLI, we evaluate policies through physical robot trials and report success rate as $100c/n$, where c is the number of successful trials and n is the total number of trials for a method and task. Completion criteria, the Candy Picking color-to-region mapping, trial counts, and confidence intervals are summarized in Appendix B and Appendix C.2.

Implementation. Our implementation builds on the FluxVLA Engine codebase (Li et al., 2026). HWAM uses a frozen Wan2.2 video VAE (Team Wan et al., 2025; Team Wan, 2025), a WanVideoDiT video expert, an ActionDiT state–action expert, and UMT5-XXL language embeddings (Chung et al., 2023). For the OLI experiments, both experts are fine-tuned for six epochs with AdamW at a learning rate of 10^{-4}, using a global batch size of 128 on 16 A800 GPUs. FDM, IDM, and Policy are sampled uniformly. ALOHA-specific settings are provided in Appendix D.

4.3 Real-Robot Evaluation

Table 1 summarizes the OLI real-robot results. The robot-data column records whether the compared model was pretrained with additional robot data as reported for the corresponding baseline.

Table 1: Real-robot evaluation results (%) on the three OLI tasks. An em dash denotes an unreported or unverified task result.

Method	Robot-data pretraining	Candy Picking	Object Collection	Plush-Toy Picking
Cosmos3-Edge	Yes	15.8	0.0	56.7
GR00T N1.5	Yes	40.5	42.5	60.0
$\pi_{0.5}$	Yes	62.5	45.0	60.7
Fast-WAM	No	43.3	0.0	60.0
DiT4DiT	No	52.5	15.0	—
HWAM (Ours)	No	**70.6**	**46.7**	**73.3**

HWAM obtains the highest observed success rate on all three OLI tasks. Relative to $\pi_{0.5}$, the observed improvements are 8.1 percentage points on Candy Picking, 1.7 points on Object Collection, and 12.6 points on Plush-Toy Picking. These results show the effectiveness of the complete HWAM policy across the evaluated stationary and mobile manipulation tasks. On Object Collection, Cosmos3-Edge and Fast-WAM fail to establish the basket grasp required by the task, accounting for their 0% full-task success rates.

4.4 Open-Loop Prediction Diagnostics

We evaluate open-loop action and state prediction on the held-out Candy Picking episodes described in Section 4.2. We use four denoising steps and compute MAE in degrees over the first 31 shared joints. Figure 5 shows MAE at each position of the 32-step chunk. Averaged over the 32 chunk positions, HWAM achieves the lowest action MAE (3.816°), compared with 4.024° for action-only FDM/IDM/Policy. Adding state supervision reduces action MAE across all evaluated training recipes. HWAM and the VGM/Policy state–action variant show similar state MAE.

For future-video prediction, we use windows immediately before grasp closure in 12 held-out episodes. Under predicted trajectory conditions, PSNR ranges from 20.345 to 20.844 and HWAM shows no video-quality advantage. Under ground-truth conditions, conditioning on both actions and post-execution states improves PSNR from 20.559 to 23.336 over the action-only model, supporting the value of realized-state conditioning. Full results are in Appendix C.1.

4.5 Controlled Ablation on Plush-Toy Picking

We evaluate four matched configurations on Plush-Toy Picking, the mobile whole-body manipulation task in the OLI suite. This task couples object handling with body movement and balance control, providing a whole-body setting to examine the interaction between the joint state–action target and the training recipe. The configurations vary two choices: the prediction target is either action-only or joint state–action, and the training recipe is either VGM/Policy or FDM/IDM/Policy. All other settings are fixed. VGM/Policy jointly optimizes future-video generation and the deployment-matched Policy

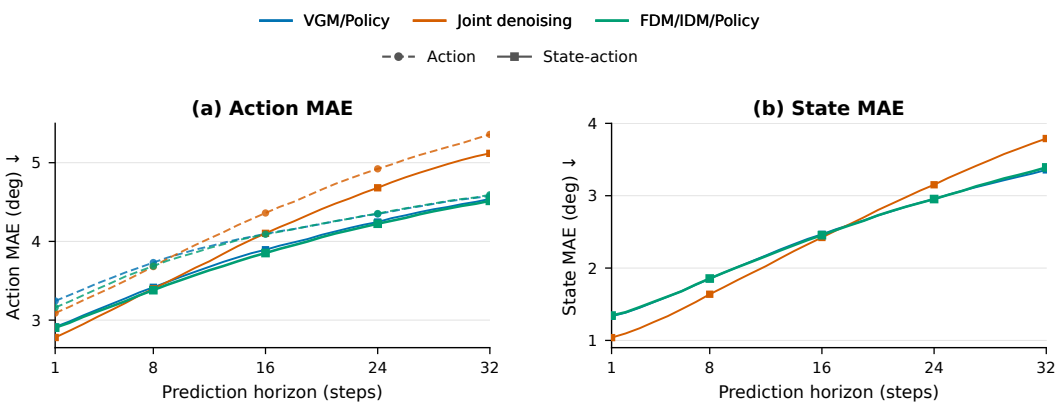

Figure 5: **Open-loop action (a) and state (b) MAE on Candy Picking over a 32-step horizon.** Errors are averaged over 31 shared joints on the held-out episodes and reported in degrees. Dashed and solid curves denote action-only and joint state–action prediction, respectively. Lower is better.

objective, whereas FDM/IDM/Policy samples separate paths for forward dynamics, inverse dynamics, and policy learning.

Table 2: Controlled ablation on Plush-Toy Picking.

Configuration	Prediction target	Training recipe	Success (%)
Action + VGM/Policy	Action	VGM/Policy	60.0
State–Action + VGM/Policy	Joint state–action	VGM/Policy	45.5
Action + FDM/IDM/Policy	Action	FDM/IDM/Policy	55.0
HWAM	Joint state–action	FDM/IDM/Policy	**73.3**

In this ablation, the effect of the joint target depends on the visual training recipe. Under VGM/Policy, adding state–action prediction decreases success from 60.0% to 45.5%. Under FDM/IDM/Policy, the same change increases success from 55.0% to 73.3%. The observed difference between these two effects is 32.8 percentage points. This target-by-recipe interaction supports combining joint state–action generation with FDM/IDM/Policy training in the evaluated setting. The complete HWAM configuration achieves the highest success rate among the four configurations. The ablation evaluates the combined formulation in this task and does not isolate the contribution of individual paths or architectural choices.

5 Conclusion

We introduced HWAM, which jointly generates high-level actions and their temporally aligned post-execution proprioceptive states. The method is motivated by a measurable action–execution gap: low-level control, dynamics, and contact can cause realized body motion to differ from policy references. The joint target is a simple extension of the WAM trajectory stream, trained with FDM and IDM to connect execution states and action references to visual transitions, alongside a Policy path that generates only state–action trajectories conditioned on current observations. HWAM achieves the highest observed success rate on the three evaluated OLI tasks. The ablation on the mobile manipulation task shows that adding the joint target improves task success under FDM/IDM/Policy training and reduces it under VGM/Policy, supporting the combined design in this setting. Additional ALOHA results demonstrate its applicability to bimanual clothes folding. HWAM does not explicitly identify low-level dynamics, and its predicted states are not sent to the controller; they serve as jointly generated variables that connect high-level actions to their physical and visual consequences.

AI USE STATEMENT

Generative AI tools were used to assist with code implementation, manuscript drafting and editing, and pre-submission feedback on the research methodology and experimental reporting, including interpretation of reported results and identification of relevant literature. The authors take responsibility for the final content of this work, including the implementation, experimental results, conclusions, and any AI-assisted text or other artifacts.

REFERENCES

Shuanghao Bai, Meng Li, Xinyuan Lv, Jiawei Wang, Xinhua Wang, Fei Liao, Chengkai Hou, Langzhe Gu, Wanqi Zhou, Kun Wu, Ziluo Ding, Zhiyuan Xu, Lei Sun, Shanghang Zhang, Zhengping Che, Jian Tang, and Badong Chen. HEX: Humanoid-Aligned Experts for Cross-Embodiment Whole-Body Manipulation. arXiv preprint arXiv:2604.07993, 2026. URL https://arxiv.org/abs/2604.07993.

Hongzhe Bi, Hengkai Tan, Shenghao Xie, Zeyuan Wang, Shuhe Huang, Haitian Liu, Ruowen Zhao, Yao Feng, Chendong Xiang, Yinze Rong, Hongyan Zhao, Hanyu Liu, Zhizhong Su, Lei Ma, Hang Su, and Jun Zhu. Motus: A Unified Latent Action World Model. arXiv preprint arXiv:2512.13030, 2025. URL https://arxiv.org/abs/2512.13030.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A Vision-Language-Action Flow Model for General Robot Control. arXiv preprint arXiv:2410.24164, 2024. URL https://arxiv.org/abs/2410.24164.

Hyung Won Chung, Xavier Garcia, Adam Roberts, Yi Tay, Orhan Firat, Sharan Narang, and Noah Constant. UniMax: Fairer and More Effective Language Sampling for Large-Scale Multilingual Pretraining. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=kXwdL1cWOAi.

Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. PaLM-E: An Embodied Multimodal Language Model. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 8469–8488. PMLR, 2023. URL https://proceedings.mlr.press/v202/driess23a.html.

Zipeng Fu, Qingqing Zhao, Qi Wu, Gordon Wetzstein, and Chelsea Finn. HumanPlus: Humanoid Shadowing and Imitation from Humans. arXiv preprint arXiv:2406.10454, 2024. URL https://arxiv.org/abs/2406.10454.

Tairan He, Zhengyi Luo, Xialin He, Wenli Xiao, Chong Zhang, Weinan Zhang, Kris M. Kitani, Changliu Liu, and Guanya Shi. OmniH2O: Universal and Dexterous Human-to-Humanoid Whole-Body Teleoperation and Learning. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 1516–1540. PMLR, 2025. URL https://proceedings.mlr.press/v270/he25b.html.

Michael Janner, Yilun Du, Joshua Tenenbaum, and Sergey Levine. Planning with Diffusion for Flexible Behavior Synthesis. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 9902–9915. PMLR, 2022. URL https://proceedings.mlr.press/v162/janner22a.html.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P Foster, Pannag R 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. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 2679–2713. PMLR, 2025. URL https://proceedings.mlr.press/v270/kim25c.html.

Moo Jin Kim, Yihuai Gao, Tsung-Yi Lin, Yen-Chen Lin, Yunhao Ge, Grace Lam, Percy Liang, Shuran Song, Ming-Yu Liu, Chelsea Finn, and Jinwei Gu. Cosmos Policy: Fine-Tuning Video Models for Visuomotor Control and Planning. arXiv preprint arXiv:2601.16163, 2026. URL https://arxiv.org/abs/2601.16163.

Yinhao Li, Weixin Mao, Zihan Lan, Jikun Rong, Qirui Hu, Yiming Zhang, Weipeng Deng, Bowen Shen, Minzhao Zhu, Yiming Mao, Yan Yang, Chenguang Cui, Hongyuan Chen, Xu Huang, Zheyi Zhao, Pinxi Shen, Bozhen He, Zhen Fu, Yifan Wang, Zexin Zhang, Ang Gao, Haoyu Chen, Chengqi Shi, and Hua Chen. FluxVLA Engine: A One-Stop VLA Engineering Platform for Embodied Intelligence. arXiv preprint arXiv:2609.17210, 2026. URL https://arxiv.org/abs/2609.17210.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow Matching for Generative Modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Teli Ma, Jia Zheng, Zifan Wang, Chunli Jiang, Andy Cui, Junwei Liang, and Shuo Yang. DiT4DiT: Jointly Modeling Video Dynamics and Actions for Generalizable Robot Control. arXiv preprint arXiv:2603.10448, 2026. URL https://arxiv.org/abs/2603.10448.

NVIDIA. GR00T-N1.5-3B: Model Card. Hugging Face model card, 2025. URL https://huggingface.co/nvidia/GR00T-N1.5-3B. Accessed September 18, 2026.

NVIDIA. Cosmos 3: Omnimodal World Models for Physical AI. arXiv preprint arXiv:2606.02800, 2026a. URL https://research.nvidia.com/labs/cosmos-lab/cosmos3/technical-report.pdf.

NVIDIA. GR00T-N1.7-3B: Model Card. Hugging Face model card, 2026b. URL https://huggingface.co/nvidia/GR00T-N1.7-3B. Early-access model card. Accessed September 18, 2026.

NVIDIA, Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi "Jim" Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, Joel Jang, Zhenyu Jiang, Jan Kautz, Kaushil Kundalia, Lawrence Lao, Zhiqi Li, Zongyu Lin, Kevin Lin, Guilin Liu, Edith Llontop, Loic Magne, Ajay Mandlekar, Avnish Narayan, Soroush Nasiriany, Scott Reed, You Liang Tan, Guanzhi Wang, Zu Wang, Jing Wang, Qi Wang, Jiannan Xiang, Yuqi Xie, Yinzhen Xu, Zhenjia Xu, Seonghyeon Ye, Zhiding Yu, Ao Zhang, Hao Zhang, Yizhou Zhao, Ruijie Zheng, and Yuke Zhu. GR00T N1: An Open Foundation Model for Generalist Humanoid Robots. arXiv preprint arXiv:2503.14734, 2025. URL https://arxiv.org/abs/2503.14734.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a Vision-Language-Action Model with Open-World Generalization. arXiv preprint arXiv:2504.16054, 2025. URL https://arxiv.org/abs/2504.16054.

Physical Intelligence, Bo Ai, Ali Amin, Raichelle Aniceto, Ashwin Balakrishna, Greg Balke, Kevin Black, George Bokinsky, Shihao Cao, Thomas Charbonnier, Vedant Choudhary, Foster Collins, Ken Conley, Grace Connors, James Darpinian, Karan Dhabalia, Maitrayee Dhaka, Jared DiCarlo, Danny Driess, Michael Equi, Adnan Esmail, Yunhao Fang, Chelsea Finn, Catherine Glossop, Thomas Godden, Ivan Goryachev, Lachlan Groom, Haroun Habeeb, Hunter Hancock, Karol Hausman, Gashon Hussein, Victor Hwang, Brian Ichter, Connor Jacobsen, Szymon Jakubczak, Rowan Jen, Tim Jones, Gregg Kammerer, Ben Katz, Liyiming Ke, Mairbek Khadikov, Chandra Kuchi, Marinda Lamb, Devin LeBlanc, Brendon LeCount, Sergey Levine, Xinyu Li, Adrian Li-Bell, Vladislav Lialin, Zhonglin Liang, Wallace Lim, Yao Lu, Enyu Luo, Vishnu Mano, Nandan Marwaha, Aikys Mongush, Liam Murphy, Suraj Nair, Tyler Patterson, Karl Pertsch, Allen Z. Ren, Gavin Schelske, Charvi Sharma, Baifeng Shi, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, Will Stoeckle, Jiaming Tang, Jimmy Tanner, Shalom Tekeste, Marcel Torne, Kyle Vedder, Quan Vuong, Anna Walling, Haohuan Wang, Jason Wang, XuDong Wang, Chris Whalen, Samuel Whitmore, Blake Williams, Charles Xu, Sukwon Yoo, Lili Yu, Wuming Zhang, Zhuoyang Zhang, and Ury Zhilinsky. $\pi_{0.7}$: a Steerable Generalist Robotic Foundation Model with Emergent Capabilities. arXiv preprint arXiv:2604.15483, 2026. URL https://arxiv.org/abs/2604.15483.

Ri-Zhao Qiu, Shiqi Yang, Xuxin Cheng, Chaitanya Chawla, Jialong Li, Tairan He, Ge Yan, David J. Yoon, Ryan Hoque, Lars Paulsen, Ge Yang, Jian Zhang, Sha Yi, Guanya Shi, and Xiaolong Wang. Humanoid Policy $\sim$ Human Policy. arXiv preprint arXiv:2503.13441, 2025. URL https://arxiv.org/abs/2503.13441.

Qiuhong Shen, Shihua Zhang, Yue Liao, Qi Li, Zhenxiong Tan, Shizun Wang, Shuicheng Yan, and Xinchao Wang. World Action Models: A Survey. arXiv preprint arXiv:2606.20781, 2026. URL https://arxiv.org/abs/2606.20781.

Mustafa Shukor, Dana Aubakirova, Francesco Capuano, Pepijn Kooijmans, Steven Palma, Adil Zouitine, Michel Aractingi, Caroline Pascal, Martino Russi, Andres Marafioti, Simon Alibert, Matthieu Cord, Thomas Wolf, and Remi Cadene. SmolVLA: A Vision-Language-Action Model for Affordable and Efficient Robotics. arXiv preprint arXiv:2506.01844, 2025. URL https://arxiv.org/abs/2506.01844.

Team Wan. Wan2.2-TI2V-5B: Model Card. Hugging Face model card, 2025. URL https://huggingface.co/Wan-AI/Wan2.2-TI2V-5B. Accessed September 18, 2026.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and Advanced Large-Scale Video Generative Models. arXiv preprint arXiv:2503.20314, 2025. URL https://arxiv.org/abs/2503.20314.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, Ayaan Malik, Kyungmin Lee, William Liang, Nadun Ranawaka, Jiasheng Gu, Yinzhen Xu, Guanzhi Wang, Fengyuan Hu, Avnish Narayan, Johan Bjorck, Jing Wang, Gwanghyun Kim, Dantong Niu, Ruijie Zheng, Yuqi Xie, Jimmy Wu, Qi Wang, Ryan Julian, Danfei Xu, Yilun Du, Yevgen Chebotar, Scott Reed, Jan Kautz, Yuke Zhu, Linxi "Jim" Fan, and Joel Jang. World Action Models are Zero-shot Policies. arXiv preprint arXiv:2602.15922, 2026. URL https://arxiv.org/abs/2602.15922.

Tianyuan Yuan, Zibin Dong, Yicheng Liu, and Hang Zhao. Fast-WAM: Do World Action Models Need Test-time Future Imagination? arXiv preprint arXiv:2603.16666, 2026. URL https://arxiv.org/abs/2603.16666.

Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware. In *Robotics: Science and Systems XIX*, 2023. URL https://roboticsproceedings.org/rss19/p016.html.

Chuning Zhu, Raymond Yu, Siyuan Feng, Benjamin Burchfiel, Paarth Shah, and Abhishek Gupta. Unified World Models: Coupling Video and Action Diffusion for Pretraining on Large Robotic Datasets. In *Proceedings of Robotics: Science and Systems*, 2025. doi: 10.15607/RSS.2025.XXI.015. URL https://roboticsproceedings.org/rss21/p015.html.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, Quan Vuong, Vincent Vanhoucke, Huong Tran, Radu Soricut, Anikait Singh, Jaspiar Singh, Pierre Sermanet, Pannag R. Sanketi, Grecia Salazar, Michael S. Ryoo, Krista Reymann, Kanishka Rao, Karl Pertsch, Igor Mordatch, Henryk Michalewski, Yao Lu, Sergey Levine, Lisa Lee, Tsang-Wei Edward Lee, Isabel Leal, Yuheng Kuang, Dmitry Kalashnikov, Ryan Julian, Nikhil J. Joshi, Alex Irpan, Brian Ichter, Jasmine Hsu, Alexander Herzog, Karol Hausman, Keerthana Gopalakrishnan, Chuyuan Fu, Pete Florence, Chelsea Finn, Kumar Avinava Dubey, Danny Driess, Tianli Ding, Krzysztof Marcin Choromanski, Xi Chen, Yevgen Chebotar, Justice Carbajal, Noah Brown, Anthony Brohan, Montserrat Gonzalez Arenas, and Kehang Han. RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control. In *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 2165–2183. PMLR, 2023. URL https://proceedings.mlr.press/v229/zitkovich23a.html.

A Action–Execution Gap Analysis

We compare each commanded joint reference a_t with its aligned post-execution proprioceptive state s_{t+1} on the LimX OLI mobile humanoid and the stationary ALOHA bimanual platform. Table 3 summarizes the gap. Both settings show non-zero action–state errors, including residual tracking errors when the commanded joint references are approximately constant. These data motivate modeling the post-execution state rather than assuming exact tracking.

Table 3: Action–execution gap measured by mean absolute error (MAE) in degrees. Direct MAE compares the joint reference a_t with the measured post-execution joint state s_{t+1}. Steady-state MAE uses the same pairing under approximately constant joint references.

Platform	Direct MAE	Steady-state MAE
LimX OLI (mobile humanoid)	4.420	2.885
ALOHA (stationary bimanual)	0.648	0.402

B Task Suite and Data

Table 4 summarizes the three OLI tasks used in the main evaluation. The episode counts refer to the robot demonstrations used for training each task.

Table 4: OLI task suite, training data, and completion criteria.

Task	Control demands	Training episodes	Completion criterion
Candy Picking	Precise grasping and spatially targeted placement	1,783	Pick the instructed candy and place it in its assigned tray region.
Object Collection	Repeated reaching and grasping across objects	4,220	Place all four tabletop objects in the basket.
Plush-Toy Picking	Mobile object handling with whole-body coordination	404	Complete the specified plush-toy handling sequence.

Training data and observations. For Candy Picking, the training set contains 1,783 demonstrations covering the three instructed candy colors. A separate set of 422 held-out episodes is used for offline validation and is excluded from training. For Object Collection and Plush-Toy Picking, the training sets contain 4,220 and 404 episodes, respectively. The OLI policy receives synchronized RGB observations from the head and wrist cameras, a language instruction, and the current proprioceptive state. Task success rates are measured in physical robot trials; the held-out demonstrations are used for prediction diagnostics.

C Additional OLI Results

C.1 Future-Video Prediction Diagnostics

We evaluate deployment-matched future-video prediction on 12 held-out episodes from the same Candy Picking task. The evaluation window is immediately before grasp closure. All conditions in Table 5 are predicted rather than taken from the ground-truth future trajectory. Here, *joint denoising* denotes joint video–trajectory denoising: the baseline corrupts and denoises both streams within one training mode, without separating the directional FDM and IDM paths.

HWAM does not obtain a video-quality advantage in this end-to-end comparison. PSNR ranges from 20.345 to 20.844 across configurations, with HWAM at 20.501. These measurements characterize future-video prediction and do not explain the differences in closed-loop task performance.

Table 5: End-to-end future-video prediction without ground-truth future state or action conditions. Higher is better.

Configuration	PSNR	SSIM
VGM + Action	20.690	0.7420
VGM + State–Action	20.673	0.7409
Joint denoising + Action	20.844	0.7453
Joint denoising + State–Action	20.670	0.7440
FDM/IDM/Policy + Action	20.345	0.7361
HWAM	20.501	0.7370

Table 6 compares the action-only FDM/IDM/Policy model and HWAM under ground-truth trajectory conditions.

Table 6: Future-video prediction with ground-truth trajectory conditions on held-out Candy Picking episodes. Higher is better.

Model	Condition	PSNR
Action-only	Ground-truth actions	20.559
HWAM	Ground-truth states and actions	23.336

HWAM achieves a PSNR of 23.336, compared with 20.559 for the action-only model. This result supports the value of realized-state conditioning for predicting visual outcomes, consistent with the action–execution gap between policy references and physical motion.

C.2 Trial Counts and Uncertainty

Table 7 supplements the Candy Picking result with available trial counts and 95% Wilson confidence intervals. These intervals assume independent Bernoulli trials and do not measure variation across training seeds.

Table 7: Candy Picking trial counts and 95% Wilson confidence intervals for methods with available counts.

Method	Successes / trials	SR (%)	95% CI (%)
GR00T N1.5	49/121	40.5	[32.2, 49.4]
$\pi_{0.5}$	75/120	62.5	[53.6, 70.6]
Fast-WAM	52/120	43.3	[34.8, 52.3]
HWAM	84/119	70.6	[61.9, 78.0]

C.3 Motion and Workspace Breakdowns

Table 8 reports available Candy Picking success rates by motion direction and workspace region. These are within-task breakdowns, not evaluations on a held-out spatial split.

Table 8: Candy Picking success rates (%) by (a) motion direction and (b) workspace region. CH/CV: cross-horizontal/vertical; X-P/X-S: primary/secondary diagonal.

(a) Motion directions				
Method	CH	CV	X-P	X-S
GR00T N1.5	50.0	25.8	40.0	46.7
$\pi_{0.5}$	60.0	66.7	80.0	43.3
Fast-WAM	33.3	43.3	46.7	50.0
HWAM	82.8	63.3	76.7	60.0

(b) Workspace regions					
Method	L1	L2	C	B2	B3
GR00T N1.5	40.0	58.3	41.7	45.8	16.7
$\pi_{0.5}$	–	–	–	–	–
Fast-WAM	70.8	83.3	41.7	8.3	12.5
HWAM	83.3	95.8	66.7	65.2	41.7

D EVALUATION ON ALOHA CLOTHES FOLDING

Task and data. We evaluate HWAM on the ALOHA bimanual platform (Zhao et al., 2023) using *Clothes Folding*, which requires coordinated grasping and successive folds of a spread garment. The task-specific training set contains 640 demonstrations. Observations comprise RGB images from a high-mounted camera and two wrist cameras, the proprioceptive state, and a language instruction. This is a task-specific evaluation on ALOHA rather than zero-shot transfer from the OLI platform. Figure 6 illustrates the task.

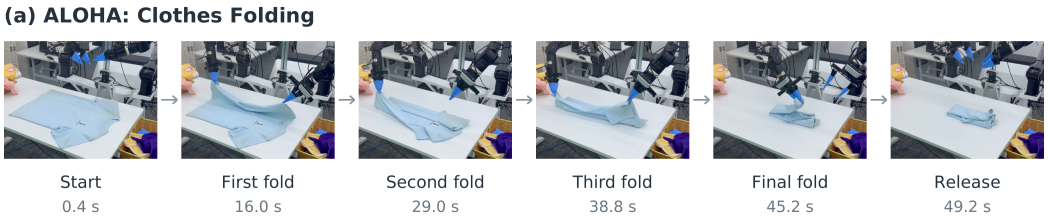

Figure 6: ALOHA Clothes Folding task. The sequence shows the robot performing successive bimanual folds on a spread garment, from the initial grasp (Start, 0.4 s) through three intermediate folds to the final fold and release (49.2 s).

Training and evaluation. The ALOHA configuration retains joint state–action prediction and uniform FDM/IDM/Policy sampling. The video and state–action experts are fine-tuned for six epochs with AdamW at a learning rate of 10^{-4} and a per-device batch size of eight, while the video VAE remains frozen. Clothes Folding is evaluated using a task-completion score on a 0–100 scale, with partial credit for completed folding stages.

Table 9: ALOHA Clothes Folding task-completion scores on a 0–100 scale. Scores include partial credit for completed folding stages. Higher is better.

Method	Clothes Folding score
GR00T N1.5	78.75
GR00T N1.7	58.75
π_0	65.00
$\pi_{0.5}$	77.00
Fast-WAM	66.25
SmolVLA	51.25
HWAM (Ours)	**86.00**

HWAM achieves the highest reported Clothes Folding task-completion score among the evaluated methods. This result supports the applicability of joint state–action generation to bimanual deformable-object manipulation beyond the humanoid platform, while the task-specific training means that it should not be interpreted as zero-shot cross-platform transfer.