HiLo-WAM: Decoupling Long-Horizon Task Evolution and Local Dynamics in World Action Models

Anonymous authors
Paper under double-blind review

Abstract

World Action Models (WAMs) improve robotic manipulation by jointly modeling visual dynamics and robot actions. However, existing WAMs primarily predict dense videos over short temporal windows, providing strong local interaction modeling but limited understanding of long-horizon task evolution. Extending visual prediction over longer horizons can provide richer task context, but also increases prediction uncertainty and computation, while forcing task-level evolution and fine-grained local dynamics to be modeled within a single temporal process. More importantly, this entangles different temporal dynamics, obscuring their dependencies and distinct roles in action generation. We propose **HiLo-WAM**, which decouples long-horizon task evolution from local visual dynamics while jointly modeling their interaction for action generation. A long-horizon visual stream captures historical task context and coarse future evolution, whereas a local visual dynamics stream models dense short-horizon robot–object interactions. The two stream operate at different temporal scales within a unified dual-DiT architecture and jointly guide robot actions through structured cross-stream interaction. Progressive stream-wise inference further reduces unnecessary world-model computation. Experiments on simulation and real-world manipulation tasks demonstrate strong long-horizon manipulation performance. HiLo-WAM achieves a 93.70% average success rate on RoboTwin 2.0 and a 17.43 long-horizon progress score on RoboDojo, while also maintaining robust performance on the Fourier GR-3 robot under both in-domain and OOD settings. Ablations further verify the complementary contributions of long-horizon task evolution and local interaction dynamics to action generation. Our project page is at https://hilowam.github.io/.

1 Introduction

Vision-Language-Action (VLA) models have achieved strong generalization in robotic manipulation through large-scale pretraining and diverse robot demonstrations Black et al. (2025; 2026a); Kim et al. (2025); Intelligence et al. (2026). World Action Models (WAMs) extend this paradigm by jointly modeling visual world dynamics and robot actions, injecting predictive world information into policy learning. They either predict future visual states to condition action generation Li et al. (2026); Hu et al. (2024); Du et al. (2023), or jointly denoise future visual representations and actions Yuan et al. (2026); Li et al. (2025a); Ma et al. (2026); Ye et al. (2026). This explicit modeling of world evolution provides a promising foundation for generalizable and long-horizon manipulation.

Most existing WAMs, however, predict dense visual dynamics over relatively short temporal windows. Such predictions provide rich supervision for fine-grained robot–object interactions, making them well suited for modeling *local interaction dynamics*. Long-horizon manipulation requires an additional capability: understanding how the task evolves beyond the immediate interaction window, including historical task progress and longer-term future development. Recent WAMs therefore introduce longer historical context, persistent memory Ye et al. (2026); Li et al. (2026); Yang et al. (2026a), extended future prediction, or long-horizon visual planning Cai et al. (2026) to incorporate longer-range task information.

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

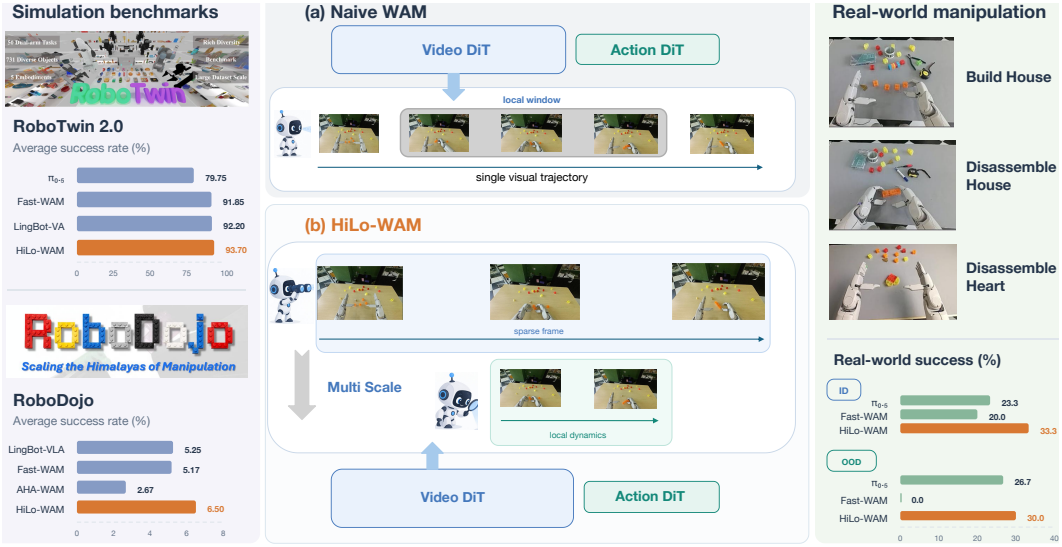

Figure 1: **Overview of HiLo-WAM.** Standard WAMs primarily model dense visual dynamics over a local interaction horizon, while long-horizon manipulation additionally requires understanding task-level evolution over extended time. HiLo-WAM decouples visual world modeling into a sparse long-horizon process for historical context and coarse future evolution and a dense local process for fine-grained interaction dynamics, and jointly models their interaction with actions.

Extending the temporal horizon of visual world modeling introduces a fundamental challenge: a single visual trajectory must capture both slowly evolving task-level progression and fine-grained local interaction dynamics. This not only increases prediction uncertainty and redundant computation, but, more importantly, obscures the dependencies between different temporal dynamics and their distinct roles in action generation. This raises a key question: *how can a World Action Model explicitly model different temporal dynamics and their dependencies on action generation?*

To address this problem, we propose **HiLo-WAM**, which decouples the two temporal roles of visual world modeling into complementary long-horizon and local dynamics streams while jointly learning their interaction with robot actions. As shown in Fig. 1, We adopt a *multi-scale temporal modeling* strategy to represent the two streams at different temporal resolutions. Specifically, the *long-horizon visual stream* sparsely models historical task context and coarse future task evolution, capturing how the task has progressed and where it is heading, while the *local visual dynamics stream* densely models short-horizon robot–object interactions required for precise control. Both streams are processed by a shared Video DiT and interact with an Action DiT through structured cross-stream attention, allowing long-horizon task information and local dynamics to jointly support action generation. This temporal decomposition further enables progressive stream-wise inference, where visual streams terminate and are cached at different stages to reduce unnecessary world-model computation.

We evaluate HiLo-WAM in simulation and real-robot tasks. HiLo-WAM achieves state-of-the-art performance on RoboTwin 2.0 and strong performance on challenging long-horizon manipulation tasks. Controlled ablations show that long-horizon historical context, coarse future evolution, and dense local visual dynamics provide complementary benefits for action generation. Further analyses of temporal scale and cross-stream attention demonstrate that long-horizon and local visual processes capture distinct yet coordinated information for control, while progressive inference further improves computational efficiency.

Our contributions are summarized as follows:

- We propose **HiLo-WAM**, a World Action Model that decouples visual world modeling into complementary long-horizon and local dynamics processes, and jointly exploits both for robot action generation.
- We introduce a multi-scale temporal modeling strategy with structured cross-stream interaction and progressive stream-wise inference, enabling effective coordination across temporal scales while reducing unnecessary world-model computation.

- Extensive simulation and real-world experiments demonstrate strong long-horizon manipulation performance, with controlled ablations validating the effectiveness of the proposed decomposition and inference design.

2 Related Works

Video Models and World Action Models. Recent advances in video generation have enabled visual prediction to serve as a powerful dynamics prior for robot learning Black et al. (2026a); Wang et al. (2026); Hu et al. (2024). Prior works use video models to generate future observations for planning and control Hu et al. (2024); Du et al. (2023); Li et al. (2026), or jointly model future visual states and robot actions within a unified World Action Model (WAM) Yuan et al. (2026); Li et al. (2025a); Ma et al. (2026); Ye et al. (2026). While early WAMs mainly focus on short-horizon interaction dynamics, recent approaches incorporate longer context, persistent memory, or long-range future prediction and visual planning Ye et al. (2026); Yang et al. (2026a); Cai et al. (2026); Zhou et al. (2025); Team et al. (2026) to support more complex long-horizon manipulation. HiLo-WAM instead decouples long-horizon manipulation into two complementary visual streams: a long-horizon stream for long-horizon task evolution and a local stream for fine-grained interaction dynamics.

Hierarchical Policies for Long-Horizon Manipulation. A parallel line of work addresses long-horizon manipulation through hierarchical or temporally abstract policies. These methods typically use high-level language plans, subgoals, visual goals, or intermediate states to capture task-level structure, and condition a downstream policy on these representations to generate fine-grained actions Shi et al. (2025); Li et al. (2025b); Long et al. (2026); Black et al. (2025); Yang et al. (2026b); Intelligence et al. (2026); Zhao et al. (2025); Hu et al. (2026). HiLo-WAM shares the motivation of combining long-range reasoning with precise local control, but approaches it through visual world modeling rather than a separate planner–policy hierarchy. It decouples the visual dynamics of a unified WAM into long-horizon and local streams and jointly learns their interaction with robot actions.

3 Method

Overview. We introduce **HiLo-WAM**, a World Action Model that decouples visual world modeling into two complementary visual streams: a *long-horizon visual stream* which sparsely models historical task context and coarse future evolution for task-level evolution and a dense *local visual dynamics stream* for fine-grained interaction dynamics. HiLo-WAM adopts a *multi-scale temporal modeling* strategy, representing the two visual streams at different temporal scales. As illustrated in Fig. 2, the model follows a dual-DiT architecture consisting of a Video DiT for visual world modeling and an Action DiT for action stream generation. Both visual streams are jointly modeled by the shared Video DiT and interact with action stream through structured cross-stream attention, allowing long-horizon task evolution and local dynamics to jointly guide action prediction. During inference, all streams are progressively denoised: the long-horizon visual stream terminates and is cached first, followed by the local visual stream, while the action stream is refined to completion.

3.1 Problem Formulation

At control step t, the robot observes an image o_t, a proprioceptive state s_t, and a language instruction l, and predicts an action chunk $A_t = (a_t, \dots, a_{t+H-1})$. Conventional World Action Models (WAMs) augment action prediction with predictive visual dynamics by jointly modeling $p_\theta(V_t, A_t \mid o_t, s_t, l)$, where $V_t = (o_{t+1}, \dots, o_{t+T})$ denotes a future visual trajectory. For long-horizon manipulation, this formulation is extended by incorporating a longer visual history $O_t = (o_{t-M}, \dots, o_t)$ and a longer future trajectory $V_t = (o_{t+1}, \dots, o_{t+T})$ to provide richer historical context and capture task evolution over a longer temporal span, yielding $p_\theta(V_t, A_t \mid O_t, s_t, l)$, where M and T control the historical and predictive temporal horizons, respectively.

3.2 Multi-scale Temporal Modeling

HiLo-WAM decouples visual world modeling into two complementary streams using a multi-scale temporal representation. Just like the standard WAM, the local visual dynamics stream

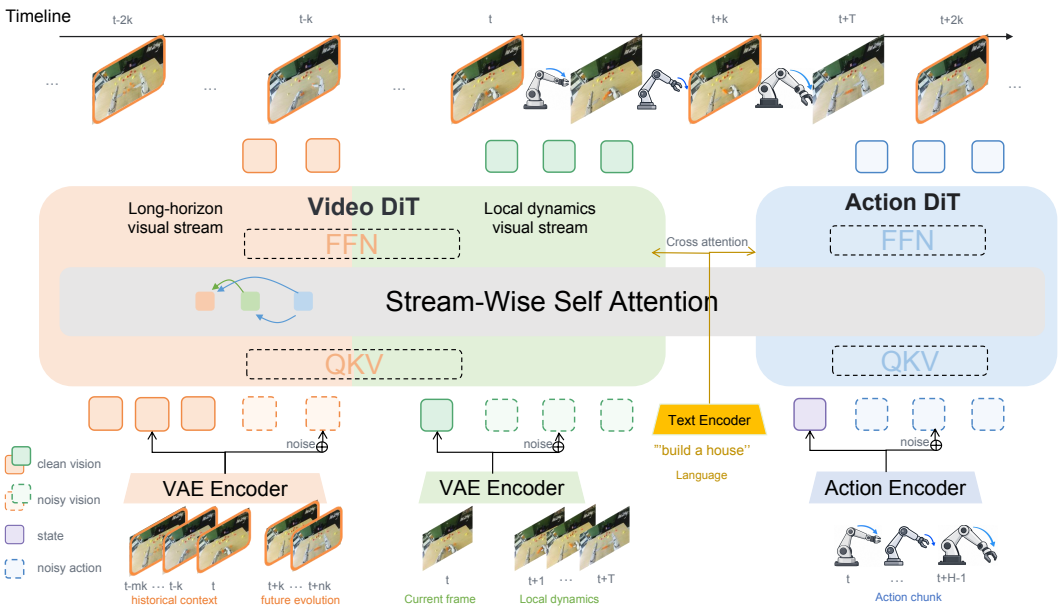

Figure 2: **Architecture of HiLo-WAM.** HiLo-WAM builds on a dual-DiT World Action Model consisting of a shared Video DiT and an Action DiT. The visual world is modeled at two complementary temporal scales: a sparse long-horizon visual stream captures historical context and coarse future evolution, while a dense local visual dynamics stream captures fine-grained local interactions. Both visual streams are jointly modeled with robot actions. During inference, the visual streams progressively terminate and are cached, while the action stream continues denoising to completion.

$V_t^{\mathrm{L}} = \{o_{t+j}\}_{j=0}^{T}$ learns to densely capture fine-grained local interactions. The long-horizon visual stream $V_t^{\mathrm{H}} = \{o_{t+ik}\}_{i=-m}^{n}$ downsamples observations with temporal stride k relative to local stream, providing broader but sparser temporal coverage for historical task context and coarse future evolution. HiLo-WAM jointly models

$$p_\theta(V_{t,i>0}^{\mathrm{H}}, V_{t,j>0}^{\mathrm{L}}, A_t \mid V_{t,i\le 0}^{\mathrm{H}}, V_{t,j=0}^{\mathrm{L}}, s_t, l), \tag{1}$$

allowing sparse historical task context and current state to predict coarse future evolution, local dynamics and action chunk.

3.3 Model Architecture

Architecture. HiLo-WAM follows the Mixture-of-Transformers (MoT) formulation of recent World Action Models Cai et al. (2026); Li et al. (2026); Yuan et al. (2026). The model consists of a Video DiT for visual world modeling and an Action DiT for robot action generation. The two branches retain modality-specific parameters and exchange information through layer-wise joint attention. We initialize the Video DiT and its video VAE from Wan2.2-5B. A pretrained T5 encoder provides language conditioning to both branches, while a action encoder maps the proprioceptive state s_t and action chunk A_t to a state token q_t and action tokens Z_t^{A} for the Action DiT.

Given the long-horizon and local visual streams defined in Sec. 3.2, we encode them independently using the same pretrained video VAE $\mathcal{E}_v$ to preserve their individual temporal structure:

$$Z_t^{\mathrm{H}} = \mathcal{E}_v(V_t^{\mathrm{H}}), \qquad Z_t^{\mathrm{L}} = \mathcal{E}_v(V_t^{\mathrm{L}}), \tag{2}$$

where Z_t^{H} and Z_t^{L} denote the long-horizon and local visual latent sequences, respectively. Finally, We organize the input tokens into four groups:

$$\mathcal{Z}_t = \left[Z_t^{\mathrm{H}}, Z_t^{\mathrm{L}}, q_t, Z_t^{\mathrm{A}}\right], \tag{3}$$

During training, latent tokens associated with history context and current observation remain clean conditions, while future visual and action tokens are noised and predicted. Information exchange among the four groups is controlled by the structured attention mechanism introduced next.

Structured cross-stream attention. After the video VAE performs spatiotemporal compression, we partition the latent video of the long-horizon stream into three parts:

$$Z_t^{\mathrm{H}} = \left[Z_t^{\mathrm{H,ctx}}, Z_t^{\mathrm{H,near}}, Z_t^{\mathrm{H,far}} \right], \tag{4}$$

where $Z_t^{\mathrm{H,ctx}}$ represents the latent historical context, $Z_t^{\mathrm{H,near}}$ corresponds to the nearest latent future frame, and $Z_t^{\mathrm{H,far}}$ represents more distant latent future frames.

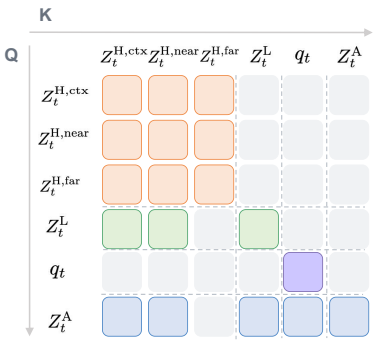

Figure 3: **Structured cross-stream attention.**

As shown in Fig. 3, HiLo-WAM employs an asymmetric cross-stream attention mask. Long-horizon latent tokens attend bidirectionally within the long-horizon stream. Local visual latents can attend to $Z_t^{\mathrm{H,ctx}}$, $Z_t^{\mathrm{H,near}}$, and the local stream itself, allowing task-level context and near-term coarse evolution to guide fine-grained dynamics modeling. Action tokens additionally attend to the proprioceptive state token and action tokens themselves, while receiving information from both visual streams.

The more distant future latents $Z_t^{\mathrm{H,far}}$ are not directly accessible to the local visual or action streams, preventing local prediction and control from relying directly on increasingly uncertain long-range visual states. Visual tokens do not attend to action tokens, while the state token q_t remains a clean condition.

3.4 Joint Flow-Matching Objective

HiLo-WAM jointly optimizes long-horizon and local dynamics visual prediction, and action generation with flow matching. For a clean target y from any prediction stream, we sample Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$ and a flow time $\tau \in [0, 1]$, and construct the interpolated sample $y_\tau = (1-\tau)y + \tau\epsilon$. The model predicts the corresponding velocity field $u = \epsilon - y$ and is trained with

$$\mathcal{L}_{\mathrm{FM}}(y) = \mathbb{E}_{y,\epsilon,\tau}\left[\| f_\theta(y_\tau, \tau, \mathcal{C}) - (\epsilon - y) \|_2^2 \right], \tag{5}$$

where $\mathcal{C}$ denotes the clean visual context, proprioceptive state, language instruction, and the accessible cross-stream context determined by the structured attention mask.

We instantiate this objective for the future portion of the long-horizon visual stream $\mathcal{L}_{\mathrm{H}} = \mathcal{L}_{\mathrm{FM}}\big(Z_{t,i>0}^{\mathrm{H}}\big)$, the future portion of the local visual stream $\mathcal{L}_{\mathrm{L}} = \mathcal{L}_{\mathrm{FM}}\big(Z_{t,j>0}^{\mathrm{L}}\big)$, and the action tokens $\mathcal{L}_{\mathrm{A}} = \mathcal{L}_{\mathrm{FM}}\big(Z_t^{\mathrm{A}}\big)$. The overall training objective is

$$\mathcal{L} = \lambda_{\mathrm{H}}\mathcal{L}_{\mathrm{H}} + \lambda_{\mathrm{L}}\mathcal{L}_{\mathrm{L}} + \lambda_{\mathrm{A}}\mathcal{L}_{\mathrm{A}}, \tag{6}$$

where λ_{H}, λ_{L}, and λ_{A} balance learning across different streams, respectively. We independently sample the flow times for the three streams, allowing them to be trained under different noise levels and supporting the progressive denoising strategy used during inference.

3.5 Efficient Progressive Inference

Fully denoising future visual predictions is computationally expensive and often unnecessary for robot control, since useful dynamics information can emerge from partially denoised visual representations Hu et al. (2024); Li et al. (2026). We therefore use stream-specific denoising budgets n_{H}, n_{L}, and n_{A} with $n_{\mathrm{H}} \le n_{\mathrm{L}} \le n_{\mathrm{A}}$ for the long-horizon, local, and action streams, respectively.

All streams are denoised jointly at first. The long-horizon stream terminates and is cached after n_{H} steps, followed by the local stream after n_{L} steps, while the action stream continues to completion. The asymmetric attention structure naturally supports this progressive partial-denoising scheme, cached upstream visual representations remain valid while downstream streams continue denoising, reducing visual-generation cost without requiring full RGB reconstruction.

4 Experiments

We conduct experiments in simulation and on a real-world Fourier GR-3 robot to investigate three questions: **(1)** Does HiLo-WAM improve long-horizon manipulation performance compared with

existing VLA and World Action Models? **(2)** How does visual dynamics at different temporal scales contribute to action generation? **(3)** How does the temporal scale affect manipulation performance?

4.1 Experimental Setup

RoboTwin 2.0. RoboTwin 2.0 is a large-scale bimanual manipulation benchmark containing 50 dual-arm tasks under both clean and domain-randomized evaluation (Chen et al., 2025). We follow the multi-task protocol of Fast-WAM (Yuan et al., 2026): the training set contains 50 clean demonstrations and 500 randomized demonstrations per task, totaling 2,500 clean and 25,000 randomized trajectories. We evaluate each task over 50 trials in both settings and report success rate averaged across all 50 tasks. The randomized setting varies scene clutter, illumination, background texture, tabletop configuration, and other visual or physical factors.

RoboDojo Benchmark. RoboDojo is a general robotic manipulation benchmark consisting of 42 tasks spanning five evaluation dimensions. We follow the official training and evaluation setup and train our model on 3,500 demonstration episodes. To assess long-horizon manipulation performance, we evaluate on eight long-horizon tasks in RoboDojo, with 50 trials per task. We report both task progress scores and success rates.

Real-world platform and tasks. We deploy policies on a Fourier GR-3 humanoid robot. We consider three long-horizon bimanual tasks: *Build House*, *Disassemble House*, *Disassemble Heart*. The assembly tasks require the policy to maintain a target structure over multiple placements, whereas the disassembly tasks require progress tracking while reversing the structure into its constituent blocks. We collect 50 hours of teleoperated demonstrations in total and train each model for five epochs on the same task-specific data.

Implementation details. HiLo-WAM is initialized from the same pretrained video backbone (Wan et al., 2025) used by the closest WAM baseline (Yuan et al., 2026). We use logit-normal flow-time sampling, AdamW with learning rate 1×10^{-4} and weight decay 0.01, a cosine learning-rate schedule, mixed-precision training, and gradient clipping at 1.0 (Yuan et al., 2026). We use 10 action denoising steps with classifier-free guidance scale 1.0. The temporal configuration (k, m, n, T, H), stream-specific stopping steps (n_H, n_L, n_A), loss weights are reported in the appendix A.

4.2 Simulation Results

Quantitative and qualitative analysis (Q1). Tables 2 and 1 summarize the results on RoboTwin 2.0 and the long-horizon tasks in RoboDojo, respectively. Overall, HiLo-WAM achieves performance comparable to state-of-the-art methods on both benchmarks without using embodied pretraining.

Model	Score	Success (%)
VLA-based methods		
OpenVLA-OFT (Kim et al., 2025)	0.70	0.00
RDT (Liu et al., 2025)	1.13	0.00
LDA-1B (Lyu et al., 2026)	1.98	0.08
H-RDT (Bi et al., 2025)	2.23	0.17
Pi-0 (Black et al., 2026a)	6.19	2.00
GROOT-N1.7 (Bjorck et al., 2025)	8.30	3.58
LingBot-VLA (Wu et al., 2026)	10.89	5.25
StarVLA (Community, 2026)	14.15	**6.50**
StarVLA-OFT	<u>14.28</u>	<u>6.17</u>
World-model-based methods		
AHA-WAM (Cai et al., 2026)	8.61	2.67
Fast-WAM (Yuan et al., 2026)	9.14	5.17
HiLo-WAM	**17.43**	**6.50**

Table 1: **RoboDojo long-horizon evaluation.** Score and average success rate (%) across 8 long-horizon tasks. Best and second-best values among the listed methods are shown in bold and underlined, respectively.

On RoboTwin, HiLo-WAM achieves the strongest result among the compared methods, with success rates of 93.87% and 93.52% in the clean and randomized settings, respectively. Its average success rate of 93.70% improves over Fast-WAM by 1.85 percentage points and over LingBot-VA by 1.50 points. The gain remains consistent under domain randomization, where HiLo-WAM exceeds Fast-WAM by 1.72 points and LingBot-VA by 2.02 points.

On RoboDojo, HiLo-WAM achieves an average score of 17.43 and a success rate of 6.50%. Its score improves over Fast-WAM by 8.29 points and over AHA-WAM by 8.82 points, while the success rate

Model	Embodied PT.	Clean	Rand.	Avg.
VLA-based methods				
π_0 (Black et al., 2026a)	✓	65.90	58.40	62.15
X-VLA (Zheng et al., 2026)	✓	72.90	72.80	72.85
$\pi_{0.5}$ (Black et al., 2025)	✓	82.70	76.80	79.75
StarVLA (Community, 2026)	×	88.20	88.30	88.25
ABot-M0 (Yang et al., 2026c)	✓	81.20	80.40	80.80
LingBot-VLA (Wu et al., 2026)	✓	86.50	85.30	85.90
World-model-based methods				
JEPA-VLA (Miao et al., 2026)	×	73.50	–	–
Motus (Bi et al., 2026)	✓	88.70	87.00	87.85
LingBot-VA (Li et al., 2026)	✓	<u>92.90</u>	91.50	<u>92.20</u>
Fast-WAM (Yuan et al., 2026)	×	91.90	<u>91.80</u>	91.85
Being-H0.7 (Luo et al., 2026)	✓	90.20	89.60	89.90
HiLo-WAM	×	**93.87**	**93.52**	**93.70**

Table 2: Average success rate (%) across 50 RoboTwin 2.0 tasks. *Embodied PT.* denotes embodied robot/action pretraining before RoboTwin training. Best and second-best values among the listed methods are shown in bold and underlined, respectively.

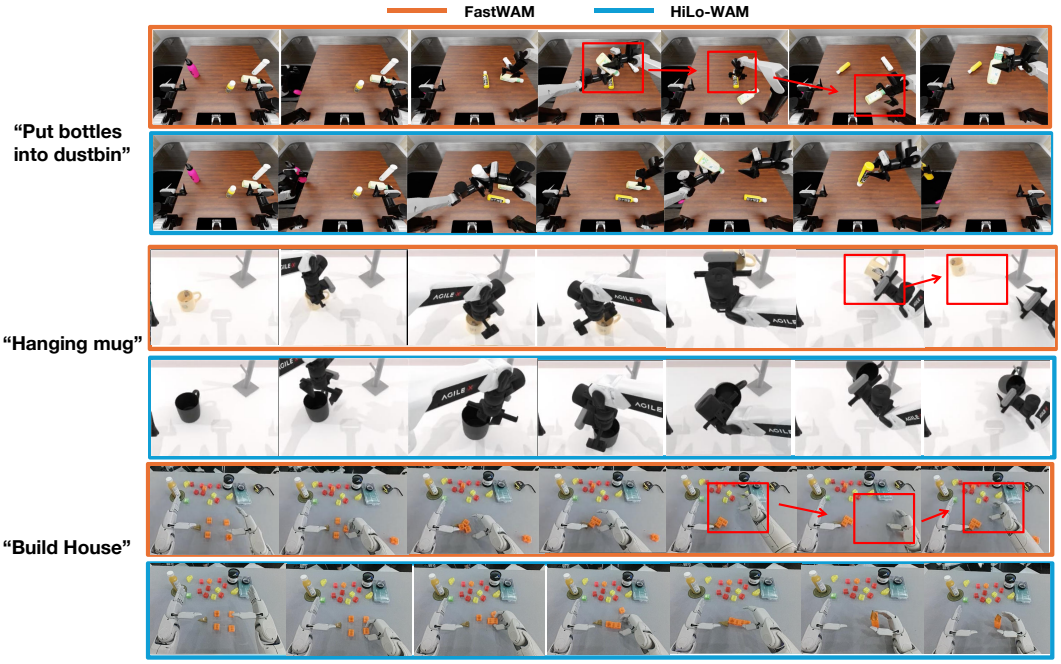

Figure 4: **Qualitative comparison between Fast-WAM and HiLo-WAM in simulation and real-world tasks.** Fast-WAM repeatedly changes its target bottle in *Put Bottles into Dustbin* and its target block in *Build House*, while HiLo-WAM maintains coherent manipulation targets throughout execution. In *Mug Hanging*, Fast-WAM fails because the mug handle is not properly aligned with the rack, whereas HiLo-WAM successfully completes the final hanging operation.

increases by 1.33 and 3.83 percentage points, demonstrating a clear advantage on challenging long-horizon manipulation tasks.

Figure 4 further shows that Fast-WAM exhibits inconsistent task progression under multimodal action distributions and fails precise local alignment in *Mug Hanging*. In contrast, HiLo-WAM maintains coherent long-horizon execution and successfully handles fine-grained local interactions, demonstrating stronger modeling of both task evolution and local dynamics.

Contribution of Multi-Scale Visual Dynamics (Q2). We next investigate how the visual dynamics at different temporal scales contribute to action generation. Table 3 ablates the historical and future components of the long-horizon visual stream together with the local visual dynamics stream.

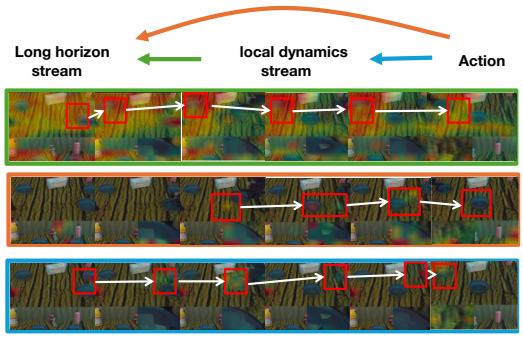

Figure 5: **Cross-stream attention map.** Colored arrows indicate unidirectional attention between streams, while red boxes highlight regions with high attention weights.

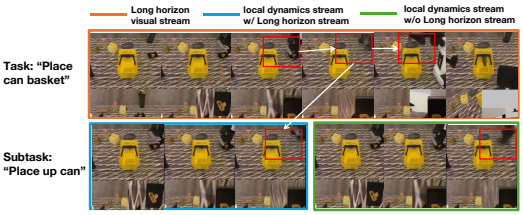

Figure 6: **Videos generated of different visual streams.** For *Place Can Basket*, the long-horizon stream predicts sparse frames over an extended horizon, whereas the local dynamics stream models the fine-grained interaction of the current subtask. Red boxes highlight the key interaction regions at the current time step.

Long-Horizon stream		Local Dynamics stream	Avg. SR
historical context	future evolution		
✗	✗	✓	90.58
✗	✓	✓	93.22
✓	✓	✗	93.52
✓	✓	✓	**93.70**

Table 3: **RoboTwin 2.0 ablations.**

Adding coarse future evolution to the local visual stream improves the average success rate from 90.58% to 93.22%, while further incorporating sparse history context increases it to 93.70%. With long-horizon visual stream enabled, removing the local visual dynamics stream reduces performance from 93.70% to 93.52%. Overall, the sparse past progress and future evolution captured by the long-horizon stream, together with fine-grained local dynamics, are all essential for action generation.

Figure 5 further reveals the mechanism by which multi-scale visual dynamics contribute to action generation. Both the action and local streams primarily attend to task-relevant objects and manipulation regions in the long-horizon stream. In contrast, action queries attend more strongly to the robot end-effector and robot–object contact regions within the local stream, reflecting the distinct roles of visual dynamics at different temporal scales in action generation. Figure 6 further shows how long-horizon context affects visual prediction. In the *Place Can Basket* task, the long-horizon stream first predicts a sparse grasp-to-place process over an extended horizon from sparse historical context. Conditioned on this prediction, the local dynamics stream generates a short-horizon grasp interaction video for the current step that is consistent with both the long-horizon plan and the overall task progression. Without long-horizon guidance, the local prediction becomes less task-directed, and the robot arm does not show a clear movement trend toward the intended manipulation. This further demonstrates the causal dependence of local interaction dynamics on long-horizon task evolution.

Effect of Temporal Scale (Q3). We analyze how different temporal scales of the long-horizon visual stream affect manipulation performance. With a fixed number of visual tokens, we vary the temporal stride k to construct long-horizon dynamics with different temporal coverage and temporal resolution.

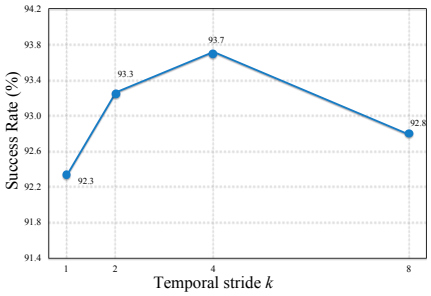

Figure 7: **RoboTwin 2.0 success rate with different temporal strides.**

As shown in Fig. 7, performance does not increase monotonically with temporal stride. Increasing k from 1 to 4 improves the success rate from 92.3% to 93.7%, suggesting that broader temporal coverage helps capture long-horizon task evolution. However, further increasing k to 8 reduces performance to 92.8%, as the visual observations become too sparse to preserve sufficient temporal information. These results indicate that long-horizon world modeling benefits from an appropriate temporal scale that balances temporal coverage and temporal resolution, with $k = 4$ achieving the best trade-off in our experiments.

Model	Build House		Disassemble House		Disassemble Heart		Average	
	ID	OOD	ID	OOD	ID	OOD	ID	OOD
$\pi_{0.5}$	2.25 / 0	1.85 / 0	8.80 / 50	**6.80 / 50**	6.60 / 20	**7.40 / 30**	5.88 / 23.3	5.35 / 26.7
Fast-WAM	2.75 / 0	1.20 / 0	7.90 / 50	3.40 / 0	4.60 / 10	3.30 / 0	5.08 / 20.0	2.63 / 0.0
HiLo-WAM	**5.65 / 10**	**4.00 / 10**	8.20 / **70**	**6.80 / 60**	**6.90 / 20**	7.30 / 20	**6.92 / 33.3**	**6.03 / 30.0**

Table 4: **Real-world evaluation on the Fourier GR-3 robot.** Each entry reports **Score / SR(%)** under in-domain (ID) and out-of-distribution (OOD) settings.

4.3 Real-World Robot Experiments

Task evaluation. We compare HiLo-WAM with $\pi_{0.5}$ (Black et al., 2025) and Fast-WAM (Yuan et al., 2026) on real-world tasks, with 10 trials per task. OOD evaluation includes changes in lighting, tabletop background, and clutter. A rollout is considered successful if the task is completed within the fixed episode horizon. To measure partial progress, each task is divided into ten ordered milestones, yielding a progress score from 0 to 10. We report the mean success rate and progress score over all trials, with task definitions and milestone criteria provided in the appendix C.

As shown in Table 4, HiLo-WAM achieves the strongest average performance in both in-domain and OOD settings. In-domain, it reaches an average score of 6.92 and a success rate of 33.3%, improving over $\pi_{0.5}$ by 1.04 score points and 10.0 percentage points in success rate, and over Fast-WAM by 1.84 points and 13.3 percentage points, respectively. Under OOD conditions, HiLo-WAM achieves an average score of 6.03 and a success rate of 30.0%, outperforming $\pi_{0.5}$ by 0.68 score points and 3.3 percentage points in success rate, while substantially improving over Fast-WAM by 3.40 score points and 30.0 percentage points. This validates the stability and robustness of HiLo-WAM on real-world long-horizon manipulation tasks.

Inference latency and real-time execution. Table 5 evaluates the inference efficiency of HiLo-WAM and the effect of progressive denoising. Multi-scale world modeling introduces additional long-horizon visual computation, leading to higher latency when all streams are fully denoised. Progressive denoising mitigates this overhead by terminating the long-horizon and local visual streams at different stages and refining only the action stream thereafter. Consequently, HiLo-WAM gains longer-horizon visual perception with only a moderate increase in inference cost. In real-world deployment, We further use Real-Time Chunking (RTC) (Black et al., 2026b) to overlap action inference with execution and reduce deployment latency.

Method	Lat. (ms)	Freq. (Hz)	Speedup
$\pi_{0.5}$	110	9.09	1.65×
Fast-WAM	182	5.49	1.00×
HiLo-WAM w/o Prog.	404	2.48	0.45×
HiLo-WAM	293	3.41	0.62×

Table 5: **Inference latency and frequency.** Speedup is measured relative to Fast-WAM.

5 Conclusion

We introduced **HiLo-WAM**, a World Action Model that separates long-horizon task evolution from fine-grained local interaction dynamics. HiLo-WAM models these two complementary processes at different temporal scales and coordinates them through structured cross-stream interaction for action generation. Progressive stream-wise denoising further limits the additional inference cost introduced by long-horizon visual modeling. Experiments in simulation and on a physical robot demonstrate strong long-horizon manipulation performance. Ablations show that long-horizon and local visual information contribute differently to action generation, while temporal-scale analysis highlights the importance of balancing temporal coverage and resolution. These results support decoupled multi-scale visual world modeling as an effective direction for long-horizon robotic manipulation.

Limitations. HiLo-WAM improves long-horizon manipulation by modeling visual dynamics at different temporal scales within the World Action Model. However, it remains limited by the semantic understanding and predictive capability of the underlying video model. Complex long-horizon tasks that require explicit subtask decomposition, progress estimation, or higher-level reasoning may therefore remain challenging. Future work could incorporate stronger semantic models, such as VLMs, to enhance task understanding, reasoning, and generalization.

AI USE STATEMENT

In this work, we did not use generative AI tools for research methodology or experimental design, theoretical development, mathematical derivations, or other scientific reasoning; these aspects were independently completed by the authors. We used generative AI tools only for language polishing and presentation refinement, including improving the clarity, fluency, and readability of author-written text. All AI-assisted content was carefully reviewed, verified, and revised by the authors. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We provide sufficient implementation and experimental details to facilitate reproduction of our results. The model architecture, training objectives, inference procedure, and key hyperparameters are described in the main paper and appendix. Detailed evaluation protocols for RoboTwin 2.0, RoboDojo, and the real-world experiments, including task definitions, OOD settings, and milestone-based progress metrics, are also provided in the appendix.

REFERENCES

Hongzhe Bi, Lingxuan Wu, Tianwei Lin, Hengkai Tan, Zhizhong Su, Hang Su, and Jun Zhu. H-rdt: Human manipulation enhanced bimanual robotic manipulation, 2025. URL https://arxiv.org/abs/2507.23523.

Hongzhe Bi, Hengkai Tan, Shenghao Xie, Zeyuan Wang, Shuhe Huang, Haitian Liu, Ruowen Zhao, Yao Feng, Chendong Xiang, Yinze Rong, et al. Motus: A unified latent action world model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 35101–35113, 2026.

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, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Robert 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. In Joseph Lim, Shuran Song, and Hae-Won Park (eds.), *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 17–40. PMLR, 27–30 Sep 2025. URL https://proceedings.mlr.press/v305/black25a.html.

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, 2026a. URL https://arxiv.org/abs/2410.24164.

Kevin Black, Manuel Galliker, and Sergey Levine. Real-time execution of action chunking flow policies. *Advances in Neural Information Processing Systems*, 38:33383–33407, 2026b.

Jisong Cai, Long Ling, Shiwei Chu, Zhongshan Liu, Jiayue Kang, Zhixuan Liang, Wenjie Xu, Yinan Mao, Weinan Zhang, Xiaokang Yang, et al. Aha-wam: Asynchronous horizon-adaptive world-action modeling with observation-guided context routing. *arXiv preprint arXiv:2606.09811*, 2026.

Tianxing Chen, Zanxin Chen, Baijun Chen, Zijian Cai, Yibin Liu, Zixuan Li, Qiwei Liang, Xianliang Lin, Yiheng Ge, Zhenyu Gu, et al. Robotwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation. *arXiv preprint arXiv:2506.18088*, 2025.

StarVLA Community. Starvla: A lego-like codebase for vision-language-action model developing, 2026. URL https://arxiv.org/abs/2604.05014.

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.

Yucheng Hu, Yanjiang Guo, Pengchao Wang, Xiaoyu Chen, Yen-Jen Wang, Jianke Zhang, Koushil Sreenath, Chaochao Lu, and Jianyu Chen. Video prediction policy: A generalist robot policy with predictive visual representations. *arXiv preprint arXiv:2412.14803*, 2024.

Yucheng Hu, Jianke Zhang, Yuanfei Luo, Yanjiang Guo, Xiaoyu Chen, Xinshu Sun, Kun Feng, Qingzhou Lu, Sheng Chen, Yangang Zhang, et al. Bagelvla: Enhancing long-horizon manipulation via interleaved vision-language-action generation. *arXiv preprint arXiv:2602.09849*, 2026.

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, 2026. URL https://arxiv.org/abs/2604.15483.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success. *arXiv preprint arXiv:2502.19645*, 2025.

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

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

Yi Li, Yuquan Deng, Jesse Zhang, Joel Jang, Marius Memmel, Caelan Garrett, Fabio Ramos, Dieter Fox, Anqi Li, Abhishek Gupta, et al. Hamster: Hierarchical action models for open-world robot manipulation. In *International Conference on Learning Representations*, volume 2025, pp. 24040–24068, 2025b.

Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. Rdt-1b: a diffusion foundation model for bimanual manipulation. In *International Conference on Learning Representations*, volume 2025, pp. 29982–30009, 2025.

Qian Long, Yueze Wang, Jiaxi Song, Junbo Zhang, Peiyan Li, Wenxuan Wang, Yuqi Wang, Haoyang Li, Shaoxuan Xie, Guocai Yao, et al. Scaling world model for hierarchical manipulation policies. *arXiv preprint arXiv:2602.10983*, 2026.

Hao Luo, Wanpeng Zhang, Yicheng Feng, Sipeng Zheng, Haiweng Xu, Chaoyi Xu, Ziheng Xi, Yuhui Fu, and Zongqing Lu. Being-h0. 7: A latent world-action model from egocentric videos. *arXiv preprint arXiv:2605.00078*, 2026.

Jiangran Lyu, Kai Liu, Xuheng Zhang, Haoran Liao, Yusen Feng, Wenxuan Zhu, Tingrui Shen, Jiayi Chen, Jiazhao Zhang, Yifei Dong, et al. Lda-1b: Scaling latent dynamics action model via universal embodied data ingestion. *arXiv preprint arXiv:2602.12215*, 2026.

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.

Shangchen Miao, Ningya Feng, Jialong Wu, Ye Lin, Xu He, Dong Li, and Mingsheng Long. Jepa-vla: Video predictive embedding is needed for vla models. *arXiv preprint arXiv:2602.11832*, 2026.

Lucy Xiaoyang Shi, Brian Ichter, Michael Equi, Liyiming Ke, Karl Pertsch, Quan Vuong, James Tanner, Anna Walling, Haohuan Wang, Niccolo Fusai, et al. Hi robot: Open-ended instruction following with hierarchical vision-language-action models. *arXiv preprint arXiv:2502.19417*, 2025.

AgiBot Research Team, Renhang Liu, Wenzhi Zhao, Zhuo Yang, Liliang Chen, Pengfei Zhou, Shengcong Chen, Guanghui Ren, Youlun Peng, Rongjun Jin, Nan Wang, Sukai Wang, Xindong He, Jinyuan Feng, Ziyu Xiong, Linqing Zhong, Yifei Wei, Feng Han, Long Zhang, Da Huang, Nanshu Zhao, Chenghao Yin, Mo Wu, Zhaodong Yan, Kongtao Hu, Yuxiang Yan, Aogelijiang Niyazi, Yu Fang, Jia Zeng, Lizhu Meng, Daizhen Lv, Haoyu Cao, Zhiwen Hou, Lianjin Ye, Yuehan Niu, Zhikai Cai, Xuan Hu, Hui Min, Xiongfeng Cai, Yue Liao, Jing Wu, Soujanya Poria, Ye Li, Sanping Zhou, and Maoqing Yao. Ge-act 2.0: Pretraining and scaling a world-action model for robotic manipulation, 2026. URL https://arxiv.org/abs/2609.05588.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Xingyu Wang, Pengxiang Ding, Jingkai Xu, Donglin Wang, and Zhaoxin Fan. Cubic: Coordinated unified bimanual perception and control framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20856–20866, June 2026.

Wei Wu, Fan Lu, Yunnan Wang, Shuai Yang, Shi Liu, Fangjing Wang, Qian Zhu, He Sun, Yong Wang, Shuailei Ma, et al. A pragmatic vla foundation model. *arXiv preprint arXiv:2601.18692*, 2026.

Sizhe Yang, Juncheng Mu, Tianming Wei, Chenhao Lu, Xiaofan Li, Linning Xu, Zhengrong Xue, Zhecheng Yuan, Dahua Lin, Jiangmiao Pang, et al. Memorywam: Efficient world action modeling with persistent memory. *arXiv preprint arXiv:2606.20562*, 2026a.

Tianshuo Yang, Guanyu Chen, Yutian Chen, Zhixuan Liang, Yitian Liu, Zanxin Chen, Chunpu Xu, Haotian Liang, Jiangmiao Pang, Yao Mu, et al. Hivla: A visual-grounded-centric hierarchical embodied manipulation system. *arXiv preprint arXiv:2604.14125*, 2026b.

Yandan Yang, Shuang Zeng, Tong Lin, Xinyuan Chang, Dekang Qi, Junjin Xiao, Haoyun Liu, Ronghan Chen, Yuzhi Chen, Dongjie Huo, et al. Abot-m0: Vla foundation model for robotic manipulation with action manifold learning. *arXiv preprint arXiv:2602.11236*, 2026c.

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.

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.

Qingqing Zhao, Yao Lu, Moo Jin Kim, Zipeng Fu, Zhuoyang Zhang, Yecheng Wu, Zhaoshuo Li, Qianli Ma, Song Han, Chelsea Finn, Ankur Handa, Tsung-Yi Lin, Gordon Wetzstein, Ming-Yu Liu, and Donglai Xiang. 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, 2025. doi: 10.1109/CVPR52734.2025.00166.

Jinliang Zheng, Jianxiong Li, Zhihao Wang, Dongxiu Liu, Xirui Kang, Yuchun Feng, Yinan Zheng, Jiayin Zou, Yilun Chen, Jia Zeng, et al. X-vla: Soft-prompted transformer as scalable cross-embodiment vision-language-action model. In *International Conference on Learning Representations*, volume 2026, pp. 60580–60606, 2026.

Pengfei Zhou, Liliang Chen, Shengcong Chen, Di Chen, Wenzhi Zhao, Rongjun Jin, Guanghui Ren, and Jianlan Luo. Act2goal: From world model to general goal-conditioned policy, 2025. URL https://arxiv.org/abs/2512.23541.

Code can be found in the Supplementary Materials.

More videos can be found at the anonymous website: `https://hilowam.github.io/`.

A Training and Implementation Details

We follow Fast-WAM Yuan et al. (2026) for data processing and backbone configuration, and summarize the additional training and inference configurations introduced for multi-scale modeling in Table 6.

Category	Parameter	Value
Model	Video backbone	Wan2.2-TI2V-5B
	Text tokenizer	Wan2.1-T2V-1.3B
	Video DiT layers	30
	Video DiT hidden / FFN dim.	3072 / 14336
	Action DiT layers	30
	Action DiT hidden / FFN dim.	1024 / 4096
	Video latent patch size	$1 \times 2 \times 2$
Data & Temporal Modeling	image resolution	240×320
	Action horizon H	32
	Video chunk T	32
	Action / video frequency ratio	4
Long-Horizon Stream	Historical keyframes m	8
	Future keyframes n	8
	Temporal stride k	4
Robot Representation	Action dimension	14
	Proprioceptive state dimension	14
	Action/state normalization	Z-score
Training	Learning rate	1×10^{-4}
	LR scheduler	Cosine
	Weight decay	1×10^{-2}
	Gradient accumulation steps	1
	Action loss weight λ_{A}	1.0
	Long-horizon stream loss weight λ_{H}	1.0
	Local dynamics stream loss weight λ_{L}	1.0
Noise Scheduler	Video train / inference shift	6.0 / 6.0
	Action train / inference shift	3.0 / 3.0
	Separated timestep conditioning	Enabled
Inference	n_H	5
	n_L	5
	n_A	10
	Joint denoising	Enabled
	Text CFG scale	1.0

Table 6: **Training and inference configurations of HiLo-WAM.** We report the main architectural, data, optimization, and inference hyperparameters used in our experiments.

B Simulation Benchmark Details

We report the per-task success rates on RoboTwin 2.0 under different temporal stride settings k, as shown in Table 7. Here, HiLo-WAM-$k2$, HiLo-WAM-$k8$ denote variants using temporal strides of $k = 2, 8$, respectively, for the long-horizon visual stream.

Task	Fast-WAM		LingBot-VA		$\pi_{0.5}$		Motus		HiLo-WAM-$k2$		HiLo-WAM-$k8$	
	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.
Adjust Bottle	**100**	**100**	90	94	**100**	99	89	93	98	**100**	94	98
Beat Block Hammer	99	97	96	**98**	96	93	95	88	**100**	96	96	92
Blocks Ranking RGB	**100**	**100**	99	98	92	85	99	97	**100**	**100**	**100**	92
Blocks Ranking Size	94	**98**	94	96	49	26	75	63	**96**	94	88	88
Click Alarmclock	**100**	**100**	99	**100**	98	89	**100**	**100**	**100**	**100**	**100**	**100**
Click Bell	**100**	**100**	**100**	**100**	99	66	**100**	**100**	**100**	**100**	**100**	**100**
Dump Bin Bigbin	97	96	89	96	92	97	95	91	98	**100**	**100**	**100**
Grab Roller	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
Handover Block	95	81	**99**	78	66	57	86	73	92	86	96	**92**
Handover Mic	99	**100**	94	96	98	97	78	63	92	**100**	**100**	98
Hanging Mug	58	**62**	40	28	18	17	38	38	46	**62**	**62**	**62**
Lift Pot	**100**	**100**	**100**	99	96	85	96	99	**100**	**100**	**100**	**100**
Move Can Pot	90	88	94	**97**	51	55	34	74	**100**	96	92	96
Move Pillbottle Pad	**100**	99	99	99	84	61	93	96	**100**	**100**	**100**	**100**
Move Playingcard Away	**100**	**100**	**100**	99	96	84	**100**	96	**100**	**100**	**100**	**100**
Move Stapler Pad	77	64	**91**	79	56	42	83	**85**	72	78	72	52
Open Laptop	**98**	**100**	92	94	90	96	95	91	**98**	98	**98**	**100**
Open Microwave	62	45	82	86	34	77	95	**91**	86	90	**100**	90
Pick Diverse Bottles	80	85	89	82	81	71	**90**	**91**	84	84	84	86
Pick Dual Bottles	**100**	96	**100**	**99**	93	63	96	90	98	94	**100**	96
Place A2B Left	95	93	**97**	93	87	82	88	79	94	**94**	90	**94**
Place A2B Right	93	**99**	**97**	95	87	84	91	87	94	96	92	94
Place Bread Basket	91	93	97	**95**	77	64	91	94	96	94	**98**	94
Place Bread Skillet	90	**93**	**95**	90	85	66	86	83	90	90	92	92
Place Burger Fries	96	99	97	95	94	87	98	98	**100**	**100**	96	98
Place Can Basket	71	69	81	**84**	62	62	81	76	**84**	82	72	78
Place Cans Plasticbox	99	96	**100**	99	94	84	98	94	**100**	**100**	98	96
Place Container Plate	96	**100**	**99**	97	**99**	95	98	99	98	98	98	**100**
Place Dual Shoes	94	88	94	89	75	75	93	87	**96**	**92**	88	90
Place Empty Cup	**100**	**100**	**100**	**100**	**100**	99	99	98	**100**	**100**	**100**	**100**
Place Fan	96	96	**99**	93	87	85	91	87	98	96	92	**100**
Place Mouse Pad	83	89	93	96	60	39	66	68	**100**	**98**	98	**98**
Place Object Basket	89	88	**91**	88	80	76	81	87	88	**90**	78	78
Place Object Scale	90	97	**96**	95	86	80	88	85	**96**	94	**96**	**98**
Place Object Stand	90	94	**99**	96	91	85	98	**97**	90	88	88	94
Place Phone Stand	97	99	97	97	81	81	87	86	94	98	**98**	**100**
Place Shoe	96	**99**	98	98	92	93	**99**	97	94	96	98	92
Press Stapler	90	97	85	82	87	83	93	**98**	**96**	**98**	**96**	94
Put Bottles Dustbin	**95**	90	87	91	84	79	81	79	92	**98**	82	92
Put Object Cabinet	**94**	89	85	87	80	79	88	71	90	88	90	**90**
Rotate QRCode	93	89	**96**	**91**	89	87	89	73	90	82	86	84
Scan Object	89	**92**	**96**	91	72	65	67	66	80	**92**	94	**92**
Shake Bottle	**100**	**100**	**100**	97	99	97	**100**	97	**100**	**100**	**100**	**100**
Shake Bottle Horizontally	**100**	**100**	**100**	99	99	99	**100**	98	**100**	**100**	**100**	**100**
Stack Blocks Three	95	97	99	98	91	76	91	95	**100**	96	98	**100**
Stack Blocks Two	**100**	**100**	**100**	98	97	**100**	**100**	98	**100**	**100**	**100**	98
Stack Bowls Three	80	81	86	83	77	71	79	87	68	80	**90**	**88**
Stack Bowls Two	92	**98**	94	**98**	95	96	**98**	**98**	92	94	88	96
Stamp Seal	90	94	96	97	79	55	93	92	**100**	96	96	**100**
Turn Switch	61	59	44	45	62	54	**84**	**78**	74	72	64	66
Average	91.88	91.78	92.90	91.50	82.74	76.76	88.66	87.02	**93.08**	**93.60**	92.76	92.76

Table 7: **Per-task success rates (%) on RoboTwin 2.0.** We report results under both clean and randomized settings. The best result for each task and setting is highlighted in bold.

C Real-World Experiment Details

We evaluate three long-horizon manipulation tasks on the physical robot: *Build House*, *Disassemble House*, and *Disassemble Heart*. Each task consists of multiple sequential manipulation stages and is assigned a progress score ranging from 0 to 10. The score is accumulated according to the completion of predefined intermediate stages, allowing us to measure partial task progress in addition to binary task success. A score of 10 corresponds to full task completion. The detailed scoring criteria are described below.

Build House. The robot is instructed to arrange four orange blocks into a straight line, connect a brown triangular block to the assembled line, lift the complete structure, and finally place it back on the table. The total score is divided into four stages:

- **Forming the block line (4 points).** Up to 4 points are awarded according to the completion and quality of the straight-line arrangement of the four orange blocks. Correctly connected and aligned blocks receive full credit, while connected but visibly bent arrangements receive partial credit.
- **Connecting the triangular block (2 points).** One point is awarded when the triangular block is moved to the designated end of the line with the correct orientation, and an additional point is awarded once a stable connection is established.
- **Lifting the assembled structure (2 points).** One point is awarded for establishing a controlled grasp on the assembled structure. The remaining point is awarded when the complete structure is lifted from the table and remains intact during the lift.
- **Replacing the structure (2 points).** One point is awarded when the robot places the structure back on the table and releases it in a controlled manner. An additional point is awarded if the structure remains connected and stable for at least two seconds after release.

Disassemble House. The robot first grasps the orange block, transfers it between the two hands, disassembles the connected structure into individual components, and places the separated pieces into the designated left and right regions of the table. The scoring criteria are:

- **Disassembling the structure (6 points).** The score is assigned proportionally to the number of required connections that are successfully separated. Full separation of all components yields 6 points. For example, when three connections must be broken, each successfully separated connection contributes 2 points.
- **Placement into left and right regions (2 points).** One point is awarded when both sides contain at least one separated block, and the second point is awarded when all components are placed inside their predefined target regions.
- **Final state (2 points).** Full credit is awarded when all components remain mutually separated and stationary for at least two seconds, with no block leaving the designated workspace.

Disassemble Heart. The robot is required to first remove the two yellow blocks from the red interlocking structure and place them on the table, and then disassemble the remaining red structure into individual components. The scoring criteria are:

- **Removing the yellow blocks (2 points).** Each yellow block contributes 1 point when it is successfully removed and stably placed on the table. A block that is removed but unintentionally dropped receives partial credit of 0.5 points.
- **Disassembling the red structure (6 points).** The score is proportional to the number of required connections that are successfully separated. Completely separating all red blocks yields 6 points. For example, when three connections are present, each separated connection contributes 2 points.
- **Final state (2 points).** Full credit is awarded when the robot has released the objects and all yellow and red blocks remain stable for at least two seconds without any object leaving the workspace.