G0.7: Scaling Up a Fast World Action Model with Long Context

Galaxea Team
https://www.galaxea-ai.com/zh/model-g07.html

Abstract

World action models (WAMs) adapt pretrained video generators for robot control by jointly predicting future camera frames and actions, transferring physical priors learned from video into manipulation policies. However, generating future frames before every action makes test-time inference too slow for closed-loop control, while most existing policies remain restricted to short observation histories. Building on Fast-WAM, which showed that video prediction shapes policy representations during training so that future frame synthesis can be skipped at deployment, we present G0.7, a generalist world action model scaled along two axes: extending the context window from a single observation to up to five minutes of history, and scaling pretraining to a large-scale corpus of robot teleoperation and human egocentric (ego) video. Across public simulation suites and real-robot deployments, direct action decoding (Fast mode) matches imagine-then-act (IDM) inference from the same checkpoint while running in 60 ms per 16-step action chunk on a single RTX 5090 (7.6× faster than IDM) and supporting pause-free real-time chunking. Performance scales consistently with pretraining data, with human ego video driving the largest gains in out-of-distribution generalization: G0.7 achieves a 70.8 mean score across five simulation benchmark suites, 48% and 54% zero-shot pick-and-place success on Galaxea R1 Pro and R1 Lite, 96.7% in-distribution and 93.3% out-of-distribution success on video-conditioned in-context learning tasks, and 86.9% mean task progress across seven real-world post-training tasks including multi-minute box folding, drink preparation, and product bagging.

1 Introduction

Generalist robot manipulation requires policies that combine physical commonsense, low-latency closed-loop reactivity, and temporal memory across extended horizons. While vision-language-action models (VLAs) [1, 19] initialize policies from pretrained vision-language models to inherit semantic reasoning, world action models (WAMs) [9, 15, 21, 26, 27] build on pretrained video generators and jointly predict future camera frames alongside robot actions, transferring rich physical and spatiotemporal priors from video into control.

Despite their promise, existing WAMs face three practical limitations when deployed on physical robots. First, most WAMs synthesize future video frames at every control step before decoding actions [9, 15, 26], incurring hundreds of milliseconds of latency per chunk and stalling closed-loop execution. Second, like most visuomotor policies, they typically condition on only one or a few recent frames, leaving the policy unable to track state over long horizons under partial observability or adapt to new tasks using human demonstrations as context. Third, publicly documented WAM pretraining corpora remain relatively modest in scale compared to the visual diversity required for zero-shot and out-of-distribution (OOD) generalization across tasks and embodiments.

In Fast-WAM [27], we addressed the first limitation by showing that test-time video generation is not what makes WAMs effective: video prediction performs its role during training by shaping the backbone's spatiotemporal representations, allowing the policy to skip future frame generation at test

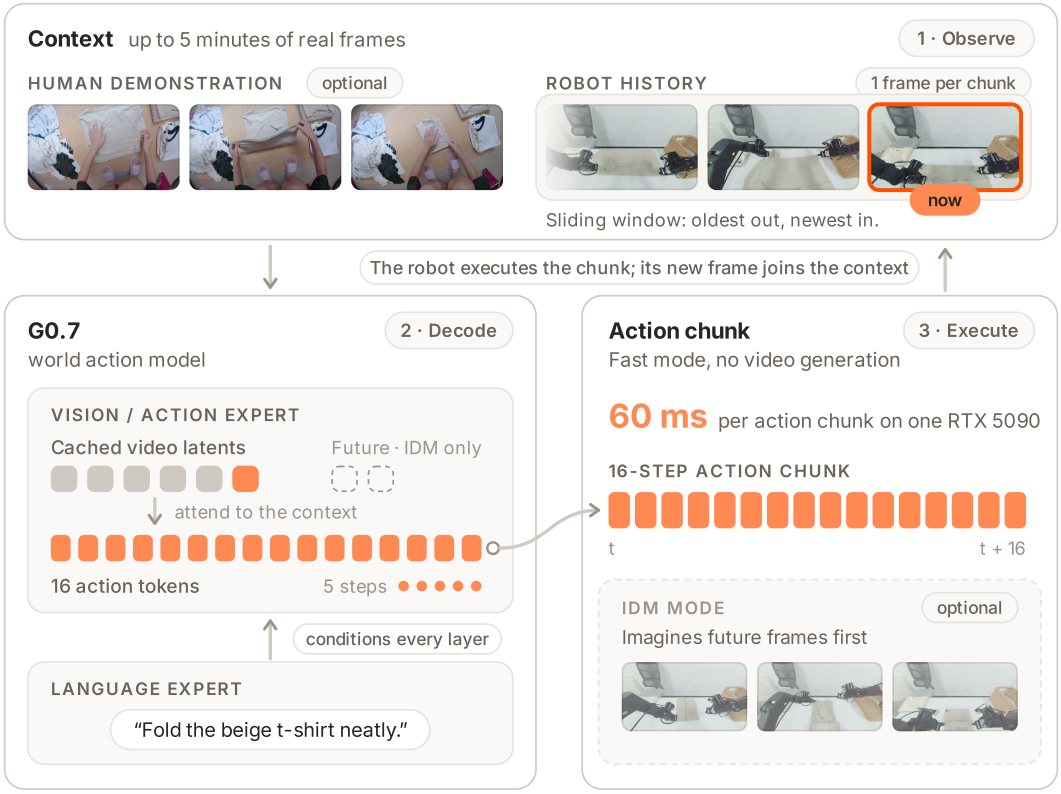


Figure 1: G0.7 architecture and control loop. The context caches up to five minutes of real camera observations, combining the robot's own history with an optional demonstration video. Conditioned at every layer by the language expert, the vision/action expert attends over this context to decode the next 16-step action chunk directly in 60 ms (Fast mode), or optionally imagines future frames first (IDM mode). After each chunk executes, the newest frame joins the context and the oldest slides out of the window.

time and decode actions directly without losing control performance. However, Fast-WAM was a small-scale study restricted to single-frame observations and benchmark-specific datasets.

In this report, we introduce G0.7, Galaxea's new generalist robot policy, which scales the Fast-WAM paradigm along both temporal context and pretraining data (Figure 1):

- **Native long-context autoregressive modeling (Section 2.1).** G0.7 conditions on up to five minutes of real camera observations maintained as clean video latents in a sliding-window KV cache. Pretrained autoregressively over multi-chunk sequences and extended during downstream adaptation without auxiliary memory modules, G0.7 supports both long-horizon episode memory and video-based in-context learning (ICL), where the model can follow a human demonstration provided as an in-context video prompt.
- **Fast path at scale (Section 2.2).** Through stochastic attention masking during training, each G0.7 checkpoint supports both imagine-then-act (IDM) and direct action decoding (Fast) from the same weights. At the full pretraining scale, Fast mode matches IDM across all four evaluation suites while running in 60 ms per 16-step action chunk in FP8 on a single RTX 5090 (156 ms vs. 457 ms in unoptimized BF16 eager mode) and integrating directly with real-time chunking (RTC) [2] for continuous motion.
- **Large-scale robot and egocentric human pretraining (Section 3).** We pretrain G0.7 on a large-scale corpus of manipulation data with multi-embodiment robot teleoperation and first-person human egocentric videos. Our data pipeline pairs three-level hierarchical language annotations and motion quality filtering on robot trajectories with 3D hand-pose extraction on human video, mapping human wrists and fingertips into a unified end-effector action representation aligned with the robot gripper convention.

Extensive evaluations across five public simulation benchmark suites and three real-robot settings (Section 4) demonstrate that G0.7 outperforms leading VLA and WAM baselines, including $\pi_{0.5}$ [19], OpenWAM [21], G0.5 [17], Dream-1 [7], and Long-WAM [11], across simulation, zero-shot deployment, in-context learning, and real-world post-training on dexterous tool use and multi-minute production workflows.

2 G0.7

G0.7 builds on a standard dual-expert world action model architecture (Figure 1). A language expert encodes the task instruction and visual context, conditioning at every layer a shared vision/action expert that operates over causal video VAE latents, proprioceptive states, and 16-step action chunks (4:1 action-to-video frequency ratio). Multi-embodiment robot trajectories and human hand trajectories are mapped into a unified state/action vector with per-embodiment dimension masking, and trained over chunk sequences under a standard joint video-action conditional flow-matching objective [27] with chunk-causal attention over clean historical video latents. Below, we focus on the two design choices that distinguish G0.7 at scale: native long-context autoregressive modeling (Section 2.1) and dual-mode Fast/IDM training and inference (Section 2.2).

2.1 Long-Context Autoregressive Modeling

While most existing VLAs and WAMs condition on only one or two recent frames, G0.7 conditions on up to five minutes of observation history, maintained as clean video latents in a sliding-window KV cache. Long-context conditioning is built into pretraining from the start: G0.7 is trained autoregressively over multi-chunk sequences with variable sequence lengths K (with per-rank batch sizes balanced to equalize compute across short and long sequences), together with random positional chunk offsets so the rotary position embeddings generalize across extended rollouts. Because attention operates over a standard sliding KV cache of clean observation latents, the context window can be further extended during downstream adaptation without architectural changes.

Beyond remembering earlier events within an episode, such as tracking an object after it is occluded, a five-minute context window enables in-context learning (ICL). Prepending a video of a person performing a task into the KV cache specifies the goal by example, allowing the robot to infer the underlying rule and continue the task autonomously. In sorting by example (Figure 2), the current camera view shows seven objects and three bins but gives no indication of which item to pick next or where it belongs; keeping the human demonstration in the KV cache resolves both decisions.

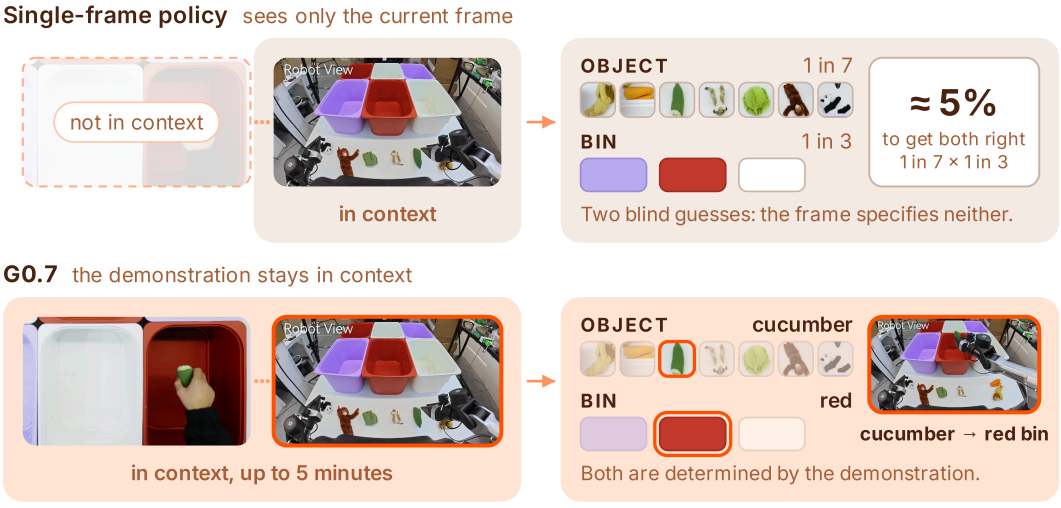


Figure 2: In-context task specification on sorting by example. From the current observation alone, the robot would have to guess both which of the seven objects to pick and which bin to put it in; attending to the human demonstration in context resolves both.

2.2 Fast Mode: Action Decoding Without Video Generation

Every G0.7 checkpoint supports two inference paths from the same weights. During teacher-forced training, we apply per-sample stochastic attention masking ($p_{\text{IDM}} = 0.5$): action tokens either attend to the current chunk's future video latents (training the IDM path) or attend only to historical and current observations (training the Fast path), while video prediction remains supervised throughout. At inference time, IDM mode first denoises future video latents and then decodes actions conditioned on them, following the standard WAM paradigm; Fast mode skips video generation altogether and denoises the 16-step action chunk directly from the cached context in five steps.

Evaluated on the same checkpoint, the two modes perform on par across all four evaluation suites of Section 4 (Figure 3). What changes is inference latency: on a single RTX 5090, Fast mode runs in 60 ms per action chunk in our deployed FP8 configuration (7.6× faster than IDM), and in 156 ms vs. 457 ms (2.9× faster) when benchmarked head-to-head in unoptimized BF16 eager mode. Video prediction thus serves its purpose as a training objective, shaping the representations from which actions are decoded, while generating video again at test time adds latency without improving control.

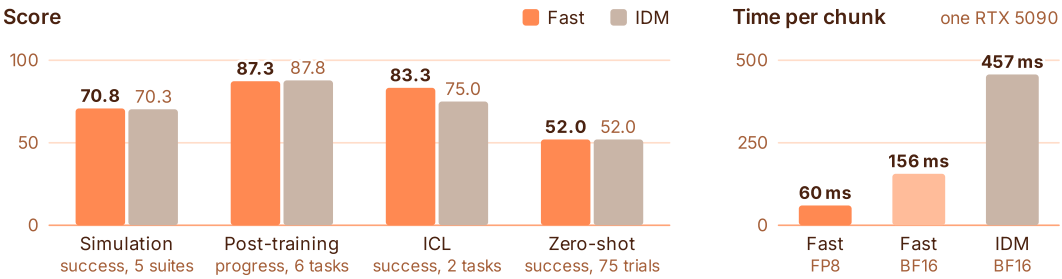


Figure 3: Fast mode vs. IDM mode from the same G0.7 checkpoint. Left: scores across the four evaluation suites of Section 4. Right: inference time per action chunk on one RTX 5090, comparing Fast mode in deployed FP8 against both modes in unoptimized BF16 eager mode.

Because Fast mode reduces inference to pure action denoising, it also makes G0.7 a drop-in fit for real-time chunking (RTC) [2]: the next action chunk is denoised while the current one executes, eliminating pauses between chunks.

3 Pretraining Data

G0.7 is pretrained on a large-scale corpus of robot teleoperation and human ego video (Figure 4). The robot corpus combines trajectories from our Galaxea R1 Pro and R1 Lite fleets with public datasets including DROID [13], RoboMIND [22], and BridgeData [20]. The ego corpus consists of first-person video of everyday manual work captured with head-mounted cameras.

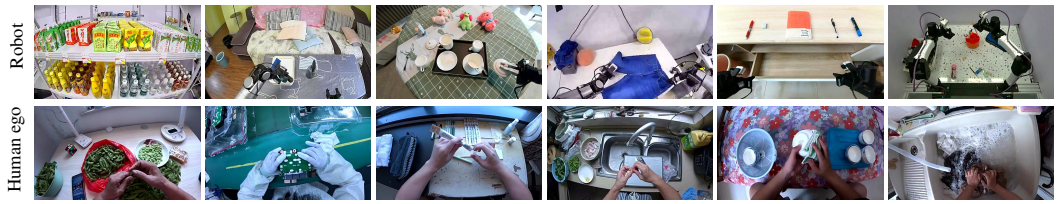


Figure 4: Sample frames from the pretraining data: robot teleoperation (top) and human ego video (bottom).

Hierarchical language annotation. Teleoperation episodes are annotated at three temporal scales (Figure 5): a complex instruction for the full episode, compound instructions for sub-goals, and atomic labels covering a few seconds of motion. Relying on episode-level instructions alone creates a supervision mismatch. Early in a shirt-folding episode, for instance, the arms are unfolding the garment while the episode label says to fold and stack it. Supervising each training window at its

matching granularity eliminates this mismatch between high-level language and low-level actions, and strengthens fine-grained instruction following so the same policy can execute both short atomic commands and multi-stage goals. Episodes also carry quality scores for motion smoothness and annotation reliability to filter out pause-heavy trajectories.

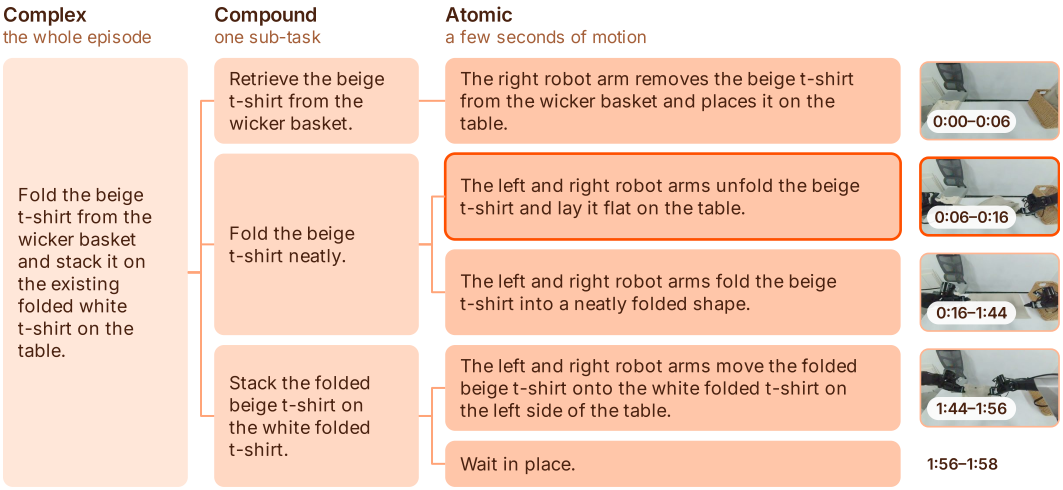


Figure 5: Three-level language annotation of a two-minute R1 Lite shirt-folding episode: one complex instruction for the full episode, three compound instructions for its sub-tasks, and five atomic labels, each paired with a representative frame and time span. Each training window is supervised at the matching temporal granularity. From 0:06 to 0:16 (outlined), the arms are unfolding the shirt, a step captured only by the atomic label.

Actions from ego video. To train on human ego video with the same objective as robot data, we built a custom data pipeline that extracts 3D hand trajectories from raw video and maps each hand into a unified end-effector action representation: the 6-DoF wrist pose aligned to the tool-center point (TCP) convention of the R1 gripper (+Z along the fingers) together with the 3D positions of all five fingertips (Figure 6). Human hands are treated as an additional embodiment under the joint video-action objective, while clips with noisy hand tracking still contribute to video prediction.



Figure 6: Extracting robot-compatible actions from human ego video. Estimated 3D hand joints in each frame are mapped to a wrist frame aligned with the R1 gripper TCP convention (+Z along the fingers) plus five fingertip positions.

4 Experiments

We evaluate G0.7 (running in Fast mode) across four settings: public simulation benchmarks, zero-shot deployment on real robots, in-context learning, and post-training on real-world manipulation tasks.

Baselines. Across these evaluations we compare against leading open-source and internal baselines: $\pi_{0.5}$ [19], the most widely adopted open-source VLA; OpenWAM [21], the state-of-the-art open-source WAM; and G0.5 [17], our previous-generation VLA. In zero-shot experiments we also include Dream-1 [7], a recent WAM from Dream Labs designed for zero-shot bimanual control, and in in-context learning we compare against Long-WAM [11], a WAM built specifically for long-horizon memory.

Checkpoints. To isolate the effect of pretraining scale and data composition, we compare three checkpoints that share the same architecture and differ only in their pretraining data (Table 1).

Table 1: Pretraining checkpoints compared in our data-scaling study.

Checkpoint	Pretraining data
G0.7-20%	About 20% of the robot teleoperation data
G0.7-Robot	The full robot corpus, without human ego video
G0.7	The full corpus, combining robot and human ego data

4.1 Simulation Benchmarks

We evaluate on five public benchmark suites that probe distinct capabilities. LIBERO [16] tests tabletop manipulation alongside three robustness suites (LIBERO-Plus [8], LIBERO-Pro [30], and LIBERO-Para [14]) that perturb scene layout, task structure, and language instructions; we report the mean of these four as LIBERO w. OOD. RoboTwin 2.0 [3] evaluates bimanual manipulation in the clean-to-random (C2R) setting under heavy domain randomization, VLABench [29] focuses on semantic instruction following, RMBench [5] tests memory-dependent manipulation, and RoboDojo [4] spans 42 challenging tasks across generalization, memory, precision, and long-horizon control. Each model is post-trained on the respective benchmark's training split and evaluated under its standard protocol.

Across the five benchmark suites, G0.7 achieves a mean score of 70.8, outperforming OpenWAM (43.9), G0.5 (39.2), and $\pi_{0.5}$ (33.9); Figure 7 shows the per-suite results. The advantage is most pronounced on RMBench, where G0.7 reaches 86.2% while single-frame baselines stay below 12%. G0.7 also leads by a wide margin on RoboTwin 2.0 (79.4% vs. 69.0% for OpenWAM) and VLABench (63.3% vs. 58.9%). On LIBERO w. OOD, G0.7 reaches 83.5% (vs. 76.9% for $\pi_{0.5}$): while standard LIBERO is near saturation across models (97%–99%), G0.7 separates itself on the perturbation suites, reaching 92.5% on LIBERO-Plus (vs. 85.7%) and 88.3% on LIBERO-Para (vs. 71.4%). On RoboDojo, G0.7 scores 41.46, more than double the strongest baseline (18.95 for G0.5).

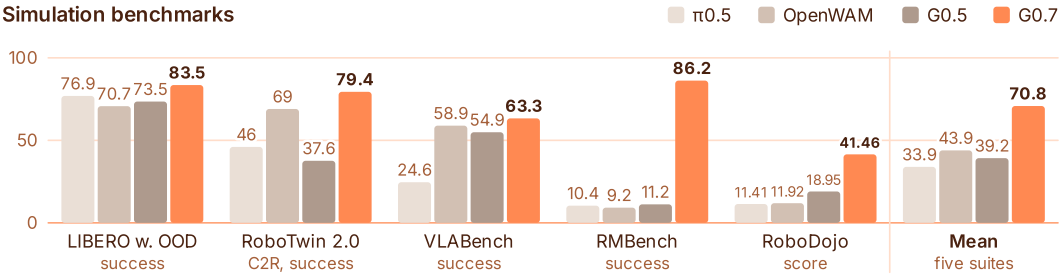


Figure 7: Simulation results after post-training on each benchmark. Scores are success rates in percent, except RoboDojo, which reports the benchmark's own score. LIBERO w. OOD is the mean of LIBERO, LIBERO-Plus, LIBERO-Pro, and LIBERO-Para.

4.2 Zero-Shot Real-Robot Deployment

To test out-of-the-box generalization without any task-specific post-training, we deploy the pretrained checkpoints directly on real-world pick-and-place (PnP) benchmarks across two Galaxea platforms: 25 trials on R1 Pro and 50 trials on R1 Lite.

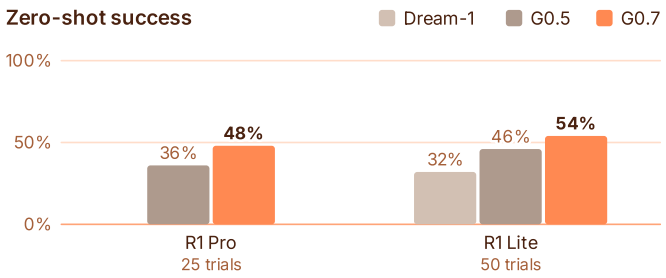


Figure 8: Zero-shot pick-and-place success rates on Galaxea R1 Pro (25 trials) and R1 Lite (50 trials). Dream-1 was evaluated on R1 Lite only.

G0.7 succeeds in 12 of 25 trials (48%) on R1 Pro and 27 of 50 trials (54%) on R1 Lite (Figure 8), outperforming both G0.5 (36% and 46%) and Dream-1 (32% on R1 Lite). Zero-shot deployment is where human ego pretraining makes the sharpest difference: without ego data, G0.7-Robot succeeds in only 6 of 50 trials (12%) on R1 Lite (Figure 13), confirming that visual and behavioral diversity from human video transfers directly to zero-shot robot control.

4.3 In-Context Learning

We next test whether G0.7 can act on information available only through extended visual contexts and generalize to unseen task configurations. We evaluate two complementary capabilities: *long-horizon visual memory* and *one-shot human demonstration following*. In the *shell game*, the robot must track hidden targets through repeated shuffling. In *sorting by example*, the robot must infer human intent from sparse cues in a single demonstration and generalize to unseen configurations (Figure 9).

Benchmark and generalization settings. Models are post-trained on task demonstrations and evaluated over 15 trials each in distribution (ID) and out of distribution (OOD). For the shell game, training and ID evaluation use three cups with 1–3 swaps, while OOD evaluation covers unseen cup types, target objects, layouts, cup counts (4–5 cups), shuffling styles, and longer sequences ($\geq$6 swaps). For sorting by example, training and ID evaluation focus on single-object sorting by demonstration, while OOD evaluation includes unseen objects, object counts, container colors and shapes, spatial layouts, and multi-object sequential sorting. Alongside standard post-training, we evaluate **ICL mid-training**, an intermediate stage between pretraining and task post-training that extends G0.7's context window and exposes the model to longer in-context demonstration sequences. We compare against Long-WAM, a competitive long-context WAM baseline, as well as G0.5 and $\pi_{0.5}$, all post-trained under aligned settings with multi-frame visual contexts provided to G0.5 and $\pi_{0.5}$.

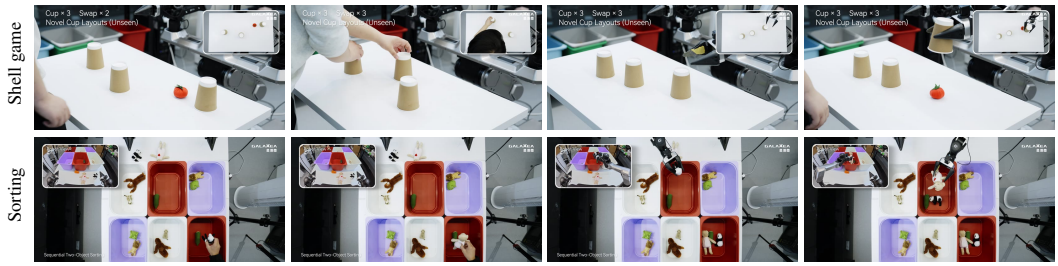


Figure 9: In-context learning tasks. *Shell game* (top): a person hides an object under a cup and shuffles the cups; G0.7 tracks the hidden target and lifts the correct cup. *Sorting by example* (bottom): a person shows where a few objects go, and G0.7 follows the human intent to sort objects into matching containers. Frames are from autonomous rollouts.

Native long-context pretraining enables stronger downstream ICL. On the shell game, G0.7 achieves 93.3% ID and 80.0% OOD, outperforming Long-WAM (53.3% / 26.7%), G0.5 (33.3% /

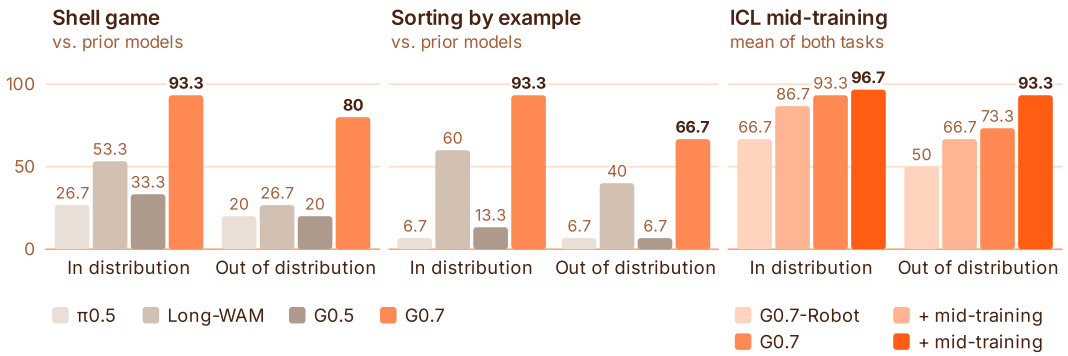


Figure 10: In-context learning success rates (%) in distribution (ID) and out of distribution (OOD), evaluated over 15 trials per setting. Left and center: comparison with prior models on the shell game and sorting by example, respectively. Right: effect of ICL mid-training on G0.7-Robot and G0.7, averaged across both tasks.

20.0%), and $\pi_{0.5}$ (26.7% / 20.0%) (Figure 10). On sorting by example, G0.7 reaches 93.3% ID and 66.7% OOD, compared with Long-WAM (60.0% / 40.0%), G0.5 (13.3% / 6.7%), and $\pi_{0.5}$ (6.7% / 6.7%). Despite receiving multi-frame visual contexts, G0.5 and $\pi_{0.5}$ remain substantially behind G0.7, highlighting the importance of learning to effectively utilize extended visual histories rather than simply providing additional context frames.

Human-to-robot transfer via egocentric pretraining. Across both post-training settings, G0.7 substantially outperforms G0.7-Robot on OOD evaluation, improving mean OOD success from 50.0% to 73.3% without mid-training and from 66.7% to 93.3% (+26.7 percentage points) with ICL mid-training. These gains suggest that egocentric human video pretraining learns transferable representations of human interactions, facilitating the transfer of human intent and task knowledge to downstream robotic manipulation.

Effect of ICL mid-training. ICL mid-training further improves mean OOD success from 50.0% to 66.7% on G0.7-Robot and from 73.3% to 93.3% on G0.7. Learning from paired human–robot demonstrations helps strengthen the correspondence between human intent and robot execution, enabling more reliable demonstration following across unseen configurations. Together with egocentric video pretraining, ICL mid-training brings G0.7 to 96.7% ID and 93.3% OOD success across both tasks.

4.4 Real-World Post-Training

Finally, we evaluate how efficiently G0.7 adapts to new physical skills across two regimes: few-shot *tool use* and *long-horizon production tasks* (Figure 11). For tool use, we collect 20 demonstrations per tool across four tasks: scooping food with a spatula, lifting a bowl with tongs, scanning a barcode with a handheld scanner, and sweeping a table with a small broom. For long-horizon production tasks, we post-train on roughly five hours of data per task to fold shipping and gift boxes from flat cardboard, make lemon tea, and bag retail products. These workflows run for several minutes and involve deformable materials and liquids, where compounding errors easily cause failure.

All models are post-trained on identical datasets and evaluated by both overall success rate and task progress (the fraction of task stages completed per trial).

Long-horizon production tasks. Across all seven evaluated post-training tasks, G0.7 achieves 86.9% mean task progress, compared with 64.3% for G0.5, 41.7% for $\pi_{0.5}$, and 18.6% for OpenWAM (Figure 12). On shipping-box folding, G0.7 achieves a 90% success rate and 95% task progress from five hours of demonstrations, compared with 75% success (79% progress) for G0.5 and 55% success (59% progress) for $\pi_{0.5}$, while OpenWAM fails to complete a full box (29.7% progress). On lemon tea, G0.7 completes every trial (100% success and task progress), compared with 80% success (88.5% progress) for G0.5 and 45% success (72% progress) for $\pi_{0.5}$, while OpenWAM never finishes the drink (33.5% progress). On product bagging, G0.7 reaches 63% success and 84% task progress,



Figure 11: Post-training tasks. Top two rows: long-horizon production tasks, each shown mid-execution and at completion, learned from roughly five hours of post-training data per task. Bottom row: the four tool-use tasks, learned from 20 demonstrations per tool. Frames are from autonomous rollouts.

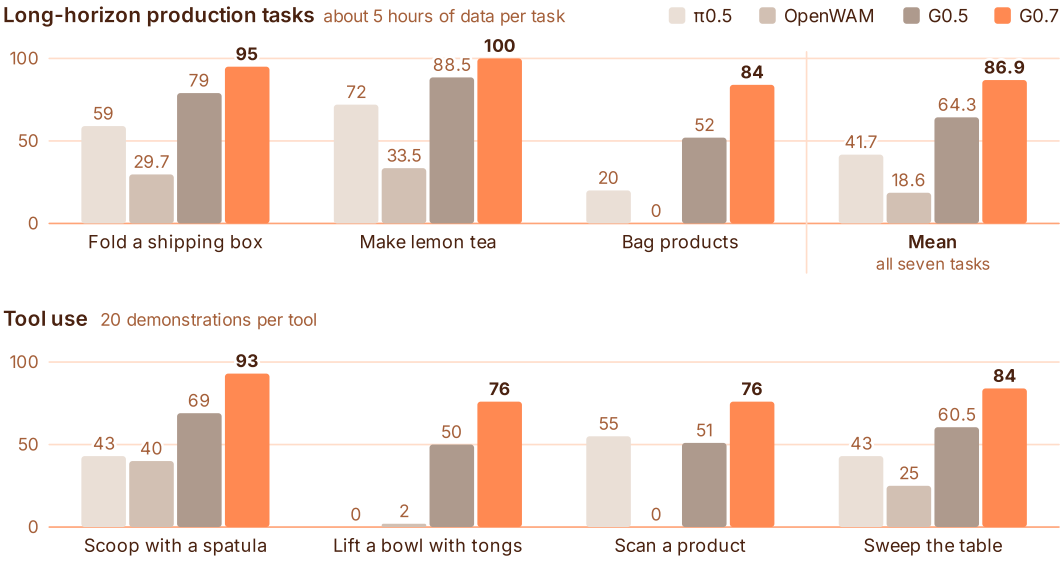


Figure 12: Task progress (%) after real-robot post-training. Top: three long-horizon production tasks (roughly five hours of data each) and the mean across all seven evaluated tasks. Bottom: four tool-use tasks (20 demonstrations per tool).

compared with 50% success (52% progress) for G0.5 and 20% success and progress for $\pi_{0.5}$, while OpenWAM makes no progress.

Few-shot tool use. From 20 demonstrations per tool, G0.7 reaches 82% mean task progress across the four tools, outperforming G0.5 (58%), $\pi_{0.5}$ (35%), and OpenWAM (17%). G0.7 maintains high progress across all four tools (93% on spatula scooping, 84% on table sweeping, and 76% on both tong grasping and barcode scanning), whereas $\pi_{0.5}$ and OpenWAM make little or no progress (0%–2%) on precision-critical tasks such as lifting a bowl with tongs.

4.5 Data Scaling

Performance improves across all four evaluation suites as pretraining scales from G0.7-20% to G0.7-Robot and G0.7 (Figure 13). Expanding the robot corpus from G0.7-20% to G0.7-Robot lifts the five-suite simulation mean from 56.2 to 66.6 and real-robot post-training progress from 81.0% to 85.9%; adding ego video in G0.7 brings them to 70.8 and 86.9%, respectively. Zero-shot deployment depends on scale the most: G0.7-20% almost never succeeds without task data (0 of 25 trials on R1 Pro and 1 of 50 on R1 Lite). Ego data has the largest impact where the policy must generalize beyond its training distribution: from G0.7-Robot to G0.7, zero-shot success on R1 Lite jumps from 12% to 54%, and out-of-distribution success on in-context learning tasks rises from 50% to 73%. Taken together, these trends reveal complementary roles for the two data sources: scaling robot teleoperation steadily improves in-distribution proficiency and downstream post-training performance, while adding human egocentric video unlocks out-of-distribution generalization across zero-shot deployment and video-conditioned in-context learning that robot data alone does not reach.

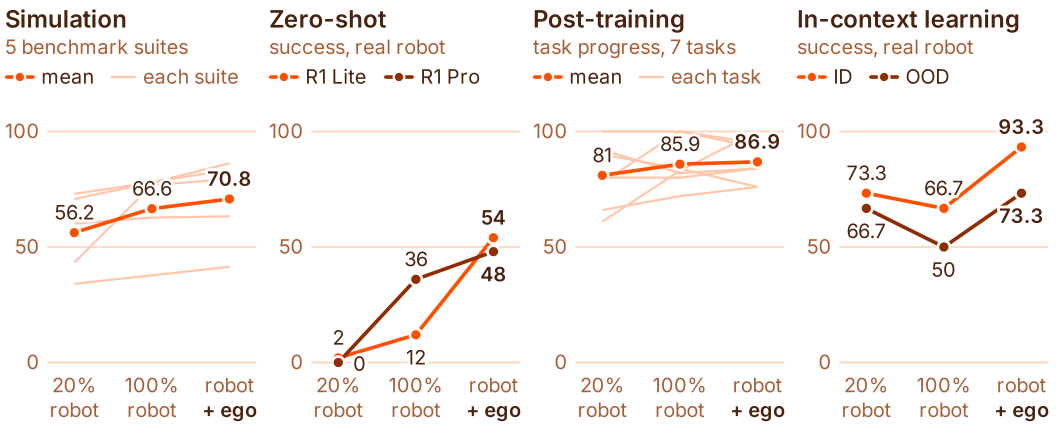


Figure 13: Scaling performance across the three pretraining checkpoints (G0.7-20%, G0.7-Robot, and G0.7) in simulation, real-robot zero-shot deployment, real-robot post-training, and in-context learning.

5 Related Work

Vision-language-action and world action models. Vision-language-action models (VLAs) such as π_0 [1] and $\pi_{0.5}$ [19] adapt pretrained vision-language backbones to output continuous robot actions via flow matching or autoregressive tokenization. World action models (WAMs) instead initialize from pretrained video generation models and co-train future video prediction alongside action decoding to transfer physical dynamics priors into control [9, 15, 21, 26–28]. Most recent WAMs, including DreamZero [26], LingBot-VA [15], and VPP2 [9], follow an imagine-then-act paradigm at deployment, denoising future video frames before predicting the corresponding actions. Fast-WAM [27] demonstrated that video supervision is crucial during representation learning but unnecessary at test time. G0.7 scales this finding to long-context autoregressive pretraining on large-scale data while retaining both direct action decoding (Fast mode) and imagine-then-act inference (IDM mode) within a single checkpoint.

Long-context, memory, and in-context robot policies. Standard visuomotor policies condition on only one or two recent observations, making them memoryless under partial observability. Recent WAMs have explored extending temporal context through persistent key-value (KV) caching across rollout chunks [15, 28], studying autoregressive context scaling [11], or introducing dedicated memory modules that compress history into anchor and gist tokens [24]. G0.7 adopts a streamlined architectural path: it maintains a sliding-window KV cache containing only clean latents from real camera observations, trains autoregressively over multi-chunk sequences from the beginning of pretraining without auxiliary memory heads, and extends its context window during downstream ICL

adaptation to follow one-shot human video demonstrations and track hidden state over multi-minute horizons.

Low-latency action chunking and asynchronous control. Predicting multi-step action chunks improves temporal consistency in high-frequency manipulation, but synchronous control loops still pause at chunk boundaries while waiting for model inference, a bottleneck that is exacerbated when WAMs must first denoise future video latents. Real-time chunking (RTC) [2] eliminates inter-chunk pauses for flow-matching policies at inference time by denoising the next chunk asynchronously while the current chunk executes, freezing committed prefix steps and inpainting the remaining horizon. Because G0.7's Fast mode bypasses video generation and reduces inference to a 60 ms flow-matching action denoising pass over cached context latents, it serves as a drop-in fit for RTC without modifying the training objective.

Learning manipulation from human egocentric video. Collecting first-person human manipulation video is significantly more scalable than robot teleoperation and captures much broader scene, object, and behavioral diversity. Prior work has leveraged human video by learning latent action spaces from action-free video [6, 25], or by estimating 3D hand and wrist trajectories and mapping them into robot action spaces [10, 12, 18, 23]. G0.7 follows the explicit 3D kinematic alignment route at scale: we extract 6-DoF wrist poses aligned to the tool-center point (TCP) axis convention of our robot grippers together with 3D five-fingertip positions, treating human hands as an additional embodiment co-trained alongside robot teleoperation under the same joint video-action objective.

6 Conclusion

G0.7 extends the Fast-WAM recipe to context windows of up to five minutes and demonstrates that direct action decoding remains effective after large-scale pretraining on robot and human ego data. With video prediction as a training objective, Fast mode performs comparably to IDM on the same checkpoint and reduces inference latency to 60 ms, enabling smooth closed-loop execution with real-time chunking. Native long-context pretraining equips the policy with both episode memory and video-based in-context learning, while the combined robot and ego corpus provides a strong foundation for zero-shot deployment and rapid adaptation to complex real-world manipulation tasks.

Contributions

Model design and pretraining. Tianyuan Yuan, Baijun Ye, Zijie Zhao, Zibin Dong, Haonan Liu, Feiyang Xue, Yicheng Liu.

Post-training and experiments. Shiduo Zhang, Xiaopeng Yu, Yizhao Xu, Baijun Ye, Zijie Zhao, Zheng Liu, Haobo Jia, Haoyu Zhang, Yang Wen.

Data. Xing Nie, Shicheng Cao, Ke Dong, Anqi Yang, Jianpeng Yu, Shenling Qiu, Haobo Jia, Tailai Cheng, Longhao Wei.

Policy infrastructure. Tianyuan Yuan, Anqi Yang, Jiahui Niu, Yue Sun.

Project lead. Tianyuan Yuan.

Project PI. Hang Zhao.

References

[1] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A Vision-Language-Action Flow Model for General Robot Control. *arXiv preprint arXiv:2410.24164*, 2024.

[2] Kevin Black, Manuel Y. Galliker, and Sergey Levine. Real-Time Execution of Action Chunking Flow Policies. *arXiv preprint arXiv:2506.07339*, 2025.

[3] 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.

[4] Tianxing Chen, Yue Chen, Zixuan Li, Junyuan Tang, Kailun Su, Haoran Lu, Weijie Wan, Baijun Chen, Songling Liu, Haowen Yan, et al. RoboDojo: A Unified Sim-and-Real Benchmark for Comprehensive Evaluation of Generalist Robot Manipulation Policies. *arXiv preprint arXiv:2607.04434*, 2026.

[5] Tianxing Chen, Yuran Wang, Mingleyang Li, Yan Qin, Hao Shi, Zixuan Li, Yifan Hu, Yingsheng Zhang, Kaixuan Wang, Yue Chen, et al. RMBench: Memory-Dependent Robotic Manipulation Benchmark with Insights into Policy Design. *arXiv preprint arXiv:2603.01229*, 2026.

[6] Yi Chen, Yuying Ge, Weiliang Tang, Yizhuo Li, Yixiao Ge, Mingyu Ding, Ying Shan, and Xihui Liu. Moto: Latent Motion Token as the Bridging Language for Learning Robot Manipulation from Videos. *arXiv preprint arXiv:2412.04445*, 2024.

[7] Dream Labs. Zero-Shot to New Robots: Introducing Dream-1. https://dreamlabs.ai/research/dream-1/, 2026.

[8] Senyu Fei, Siyin Wang, Junhao Shi, Zihao Dai, Jikun Cai, Pengfang Qian, Li Ji, Xinzhe He, Shiduo Zhang, Zhaoye Fei, et al. LIBERO-Plus: In-depth Robustness Analysis of Vision-Language-Action Models. *arXiv preprint arXiv:2510.13626*, 2025.

[9] Yanjiang Guo, Haodong Yan, Zhide Zhong, Zhongru Zhang, Qingyuan Yang, Qingzhou Lu, Xiaoyu Chen, Yen-Jen Wang, Shuying Deng, Chenghan Yang, et al. Video Prediction Policy 2: Predict Better, Act Better. *arXiv preprint arXiv:2610.10270*, 2026.

[10] Ryan Hoque, Peide Huang, David J. Yoon, Mouli Sivapurapu, and Jian Zhang. EgoDex: Learning Dexterous Manipulation from Large-Scale Egocentric Video. *arXiv preprint arXiv:2505.11709*, 2025.

[11] Wei Huang, Bohan Zhang, Chenzhi Liu, Isabella Liu, Shuai Yang, Weian Mao, Luozhou Wang, Yicheng Xiao, Weifeng Lin, Qixin Hu, et al. Long-WAM: Scaling the Context of World-Action Models. *arXiv preprint arXiv:2610.10528*, 2026.

[12] Simar Kareer, Dhruv Patel, Ryan Punamiya, Pranay Mathur, Shuo Cheng, Chen Wang, Judy Hoffman, and Danfei Xu. EgoMimic: Scaling Imitation Learning via Egocentric Video. *arXiv preprint arXiv:2410.24221*, 2024.

[13] Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. DROID: A Large-Scale In-The-Wild Robot Manipulation Dataset. *arXiv preprint arXiv:2403.12945*, 2024.

[14] Chanyoung Kim, Minwoo Kim, Minseok Kang, Hyunwoo Kim, and Dahuin Jung. LIBERO-Para: A Diagnostic Benchmark and Metrics for Paraphrase Robustness in VLA Models. *arXiv preprint arXiv:2603.28301*, 2026.

[15] Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Fei Han, Mingrui Yu, Zelin Gao, Nan Xue, Xing Zhu, Yujun Shen, and Yinghao Xu. Causal World Modeling for Robot Control. *arXiv preprint arXiv:2601.21998*, 2026.

[16] Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking Knowledge Transfer for Lifelong Robot Learning. *arXiv preprint arXiv:2306.03310*, 2023.

[17] Yicheng Liu, Zibin Dong, Baijun Ye, Tianyuan Yuan, Tao Jiang, Anqi Yang, Shicheng Cao, Haonan Liu, Yue Sun, Zihan Guo, et al. G0.5: One Autoregressive Stream for Robot Reasoning and Action. *arXiv preprint arXiv:2608.11739*, 2026.

[18] Hao Luo, Yicheng Feng, Wanpeng Zhang, Sipeng Zheng, Ye Wang, Haoqi Yuan, Jiazheng Liu, Chaoyi Xu, Qin Jin, and Zongqing Lu. Being-H0: Vision-Language-Action Pretraining from Large-Scale Human Videos. *arXiv preprint arXiv:2507.15597*, 2025.

[19] Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. $\pi_{0.5}$: a Vision-Language-Action Model with Open-World Generalization. *arXiv preprint arXiv:2504.16054*, 2025.

[20] Homer Walke, Kevin Black, Abraham Lee, Moo Jin Kim, Max Du, Chongyi Zheng, Tony Zhao, Philippe Hansen-Estruch, Quan Vuong, Andre He, et al. BridgeData V2: A Dataset for Robot Learning at Scale. *arXiv preprint arXiv:2308.12952*, 2023.

[21] Yuran Wang, Siqiao Huang, Mingleyang Li, Chenhao Zhang, Jiaqi Liang, Weiyang Jin, Yue Chen, Xuemin Chi, Donghao Zhou, Qize Yu, et al. OpenWAM: An Open, Modular Exploration Towards Systematic World-Action Model Pretraining. *arXiv preprint arXiv:2609.07398*, 2026.

[22] Kun Wu, Chengkai Hou, Jiaming Liu, Zhengping Che, Xiaozhu Ju, Zhuqin Yang, Meng Li, Yinuo Zhao, Zhiyuan Xu, Guang Yang, et al. RoboMIND: Benchmark on Multi-embodiment Intelligence Normative Data for Robot Manipulation. *arXiv preprint arXiv:2412.13877*, 2024.

[23] Ruihan Yang, Qinxi Yu, Yecheng Wu, Rui Yan, Borui Li, An-Chieh Cheng, Xueyan Zou, Yunhao Fang, Xuxin Cheng, Ri-Zhao Qiu, et al. EgoVLA: Learning Vision-Language-Action Models from Egocentric Human Videos. *arXiv preprint arXiv:2507.12440*, 2025.

[24] Sizhe Yang, Juncheng Mu, Tianming Wei, Chenhao Lu, Xiaofan Li, Linning Xu, Zhengrong Xue, Zhecheng Yuan, Dahua Lin, Jiangmiao Pang, and Huazhe Xu. MemoryWAM: Efficient World Action Modeling with Persistent Memory. *arXiv preprint arXiv:2606.20562*, 2026.

[25] Seonghyeon Ye, Joel Jang, Byeongguk Jeon, Sejune Joo, Jianwei Yang, Baolin Peng, Ajay Mandlekar, Reuben Tan, Yu-Wei Chao, Bill Yuchen Lin, et al. Latent Action Pretraining from Videos. *arXiv preprint arXiv:2410.11758*, 2024.

[26] 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.

[27] 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.

[28] Qihang Zhang, Lin Li, Luyao Zhang, Shuai Yang, Yiming Luo, Shuaiting Li, Ruilin Wang, Junke Wang, Jiahao Shao, Gangwei Xu, et al. Native Video-Action Pretraining for Generalizable Robot Control. *arXiv preprint arXiv:2607.08639*, 2026.

[29] Shiduo Zhang, Zhe Xu, Peiju Liu, Xiaopeng Yu, Yuan Li, Qinghui Gao, Zhaoye Fei, Zhangyue Yin, Zuxuan Wu, Yu-Gang Jiang, and Xipeng Qiu. VLABench: A Large-Scale Benchmark for Language-Conditioned Robotics Manipulation with Long-Horizon Reasoning Tasks. *arXiv preprint arXiv:2412.18194*, 2024.

[30] Xueyang Zhou, Yangming Xu, Guiyao Tie, Yongchao Chen, Guowen Zhang, Duanfeng Chu, Pan Zhou, and Lichao Sun. LIBERO-PRO: Towards Robust and Fair Evaluation of Vision-Language-Action Models Beyond Memorization. *arXiv preprint arXiv:2510.03827*, 2025.