VoLo: A Physical Orchestrator for Open-Vocabulary Long-Horizon Manipulation

Siyi Chen[1,2], **Hugo Hadfield**[1], **Alex Zook**[1], **Mikaela Angelina Uy**[1], **Chan Hee Song**[1], **Erwin Coumans**[1], **Xuning Yang**[1], **Faisal Ladhak**[1], **Qing Qu**[2], **Stan Birchfield**[1], **Jonathan Tremblay**[1†], **Valts Blukis**[1†]

[1]NVIDIA [2]University of Michigan [†]Project Leads

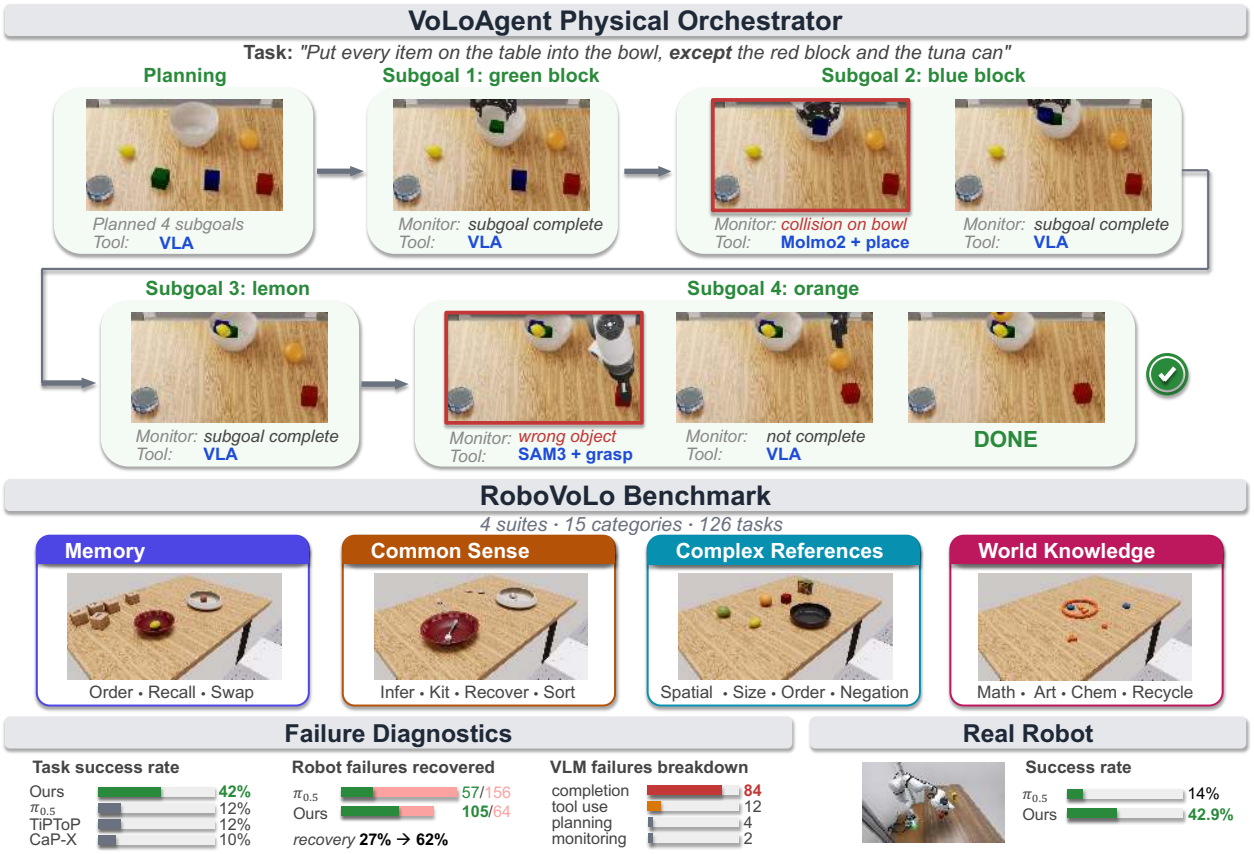

Figure 1: **VoLo overview.** VoLoAgent plans, monitors (e.g., subgoal complete), and uses tools (e.g., VLA, SAM3) to act and recover from failures (e.g., wrong object). RoboVoLo is a high-fidelity benchmark for evaluating and diagnosing open-vocabulary long-horizon manipulation.

Abstract

Open-vocabulary long-horizon manipulation requires robots to reason over flexible instructions and complex multi-object scenes while adaptively planning, executing, monitoring, and recovering from failures. We address these demands with a closed agent loop in which a VLM orchestrates heterogeneous robot capabilities as interruptible tools. Unlike in virtual AI agents, the timing of decisions, actions and tool calls is important in a physical world that does not pause for reasoning. We refer to this setting as *Physical Orchestration*, and propose VoLoAgent, a VLM that plans, monitors, and recovers by treating a VLA/WAM as an interruptible tool it steers mid-rollout alongside vision models and action primitives. To evaluate these long-horizon capabilities, we introduce RoboVoLo, a high-fidelity benchmark for open-vocabulary long-horizon manipulation across common sense, memory/state tracking, complex references, and world knowledge, with both task-level success and failure-mode diagnostics. Experiments show VoLoAgent substantially outperforms single VLA/VLM or tool-based systems, with validation on real-robot experiments. Project page: https://chicychen.github.io/VoLo/

* *Work done during an internship at NVIDIA.*

1. Introduction

Real-world manipulation is often open-vocabulary and long-horizon, rather than a template pick-and-place task. As illustrated in Fig. 1, when asked to "put every item on the table into the bowl, except the red block and the tuna can," a robot must understand negative references such as "except," plan over a sequence of objects, monitor whether each subgoal succeeds, and recover from failures such as picking the wrong object. These open-vocabulary long-horizon tasks require high-level capabilities including planning, reasoning over complex language, using world knowledge, spatial reasoning, and maintaining memory of the evolving scene. At the same time, they demand reliable embodied perception and precise low-level action skills.

Existing manipulation approaches only partially address these challenges. End-to-end vision-language-action (VLA) models (Intelligence et al., 2025,, 2026) and world action models (WAMs) (Cheang et al., 2024; Gao et al., 2026; Kim et al., 2026; Li et al., 2026; Ye et al., 2026) exhibit precise manipulation, but lack robust planning, monitoring, and perception in the multi-object scenes typical of long-horizon tasks. LLM/VLM-driven code-as-policy methods (Fu et al., 2026; Liang et al., 2023; Singh et al., 2023), including tool-augmented variants (Chen et al., 2026), support explicit reasoning over perception and classical control primitives, but are limited by fixed toolsets and control APIs for contact-rich manipulation, while largely overlooking monitoring and recovery. Recent hierarchical systems pair a VLM planner with a VLA executor (Li et al., 2025; Lin et al., 2025; Liu et al., 2026; Shi et al., 2025; Yang et al., 2025; Yi et al., 2026), but usually hard-wire this control flow rather than adaptively composing VLA/WAMs with perception, action, monitoring, and recovery tools. In short, the VLA is treated as a fixed executor rather than one interruptible capability among many.

We instead approach open-vocabulary long-horizon manipulation as *physical orchestration*: unlike a virtual agent, which can pause the world while it thinks, a physical agent must decide *when* to act, advance, or stop against a world that keeps moving (Sec. 4.1). We present VoLoAgent, an instantiation of this idea that unifies a VLA/WAM with perception models and grasp/place primitives as callable tools in a flexible VLM-managed agent loop, and outperforms hard-wired pipelines.

To study this regime, we introduce RoboVoLo, a high-fidelity benchmark for open-vocabulary long-horizon manipulation built on RoboLab (Yang et al., 2026). Existing benchmarks (Kim et al., 2026; Liu et al., 2023; Yang et al., 2026; Zhu et al., 2020) often focus on short-horizon skills, overlook open-vocabulary reasoning, or use simplified scenes, leaving limited room to study long-horizon state tracking and adaptive recovery. RoboVoLo spans four suites: common sense, memory, complex references, and world knowledge, comprising 15 task categories and 126 tasks in total. Comprehensive experiments show that VoLoAgent substantially outperforms standalone action models, code-as-policy systems, and TAMP-style baselines (*i.e.,* task and motion planning). We further analyze both robot-level failures, such as wrong-object picks and stuck behavior, and VLM-level failures, such as planning mistakes, missed failure detection, and tool-use errors, to diagnose the strengths and limitations of tool-augmented robotic agents. Finally, we validate our findings on real Franka manipulation tasks, showing that orchestration substantially improves over a standalone action model.

We make the following contributions:

1. VoLoAgent, an adaptive tool-augmented robotic agent that uses a VLM to plan, reason, monitor, and recover by composing an interruptible VLA/WAM with perception models and classical action primitives callable tools in a single closed loop.
2. RoboVoLo, a high-fidelity benchmark with 126 tasks for open-vocabulary long-horizon manipulation, spanning common sense, memory, references, and world knowledge, designed independently of the system.
3. A large-scale empirical study comparing action models, code-as-policy systems, TAMP-style systems, and ablations of VoLoAgent orchestrator, complemented by real robotic experiments.

2. Related Work

Vision-Language-Action and World Action Models. End-to-end VLAs map observations and instructions directly to robot actions, achieving strong dexterity at scale (Brohan et al., 2023; Deshpande et al., 2026; Fang et al., 2026; Intelligence et al., 2025,, 2026; Jiang et al., 2023; Kim et al., 2024; Lee et al., 2025; Liu et al., 2025); world action models (WAMs) extend this line by jointly predicting future video and actions (Cheang et al., 2024; Gao et al., 2026; Kim et al., 2026; Li et al., 2026; Ye et al., 2026). Recent variants interleave explicit reasoning, dual-system architectures, chain-of-thought planning, or depth-aware spatial tokens (Intelligence et al., 2025; Lee et al., 2025; Zhang et al., 2024), and some push memory inside the policy via memory banks Shi et al. (2026) or multi-frame chunking Li et al. (2026). However, their action chunks still execute largely open-loop, limiting planning, reasoning, real-time monitoring, and tool-based recovery during execution. We instead use VLA/WAM as an interruptible tool inside a physical orchestrator.

Agentic and Hierarchical Robot Frameworks. LLM- and VLM-driven program synthesis grounds high-level reasoning in robotic primitives via code generation (Fu et al., 2026; Liang et al., 2023; Singh et al., 2023) or closed-loop VLM verification (Huang et al., 2022; Zhi et al., 2025), with TAMP-augmented variants guiding symbolic task-and-motion planners (Shen et al., 2026; Yang et al., 2024). These remain limited by fixed primitive interfaces and largely overlook real-time monitoring and failure recovery. A parallel line stacks a VLM planner above a VLA executor (Duan et al., 2024; Lei et al., 2026; Li et al., 2025; Liu et al., 2026; Ma et al., 2026; Schakkal et al., 2025; Shi et al., 2025; Yang et al., 2026, 2025), sometimes paired with a critic for failure detection and replanning (Chen et al., 2026; Dai et al., 2024; Duan et al., 2025; Feng et al., 2025; Fu et al., 2026; Gu et al., 2025; Lin et al., 2025; Liufu et al., 2026; Mei et al., 2024; Pchelintsev et al., 2025; Skreta et al., 2024; Yang et al., 2026; Ye et al., 2025; Yi et al., 2026). Concurrent work Lei et al. (2026) routes a VLM through a family of specialized VLAs. However, these systems still treat the VLM-VLA call as a hardwired pipeline; in contrast, our physical orchestrator treats the VLA/WAM as one interruptible tool among others, enabling real-time monitoring, mid-rollout intervention, and adaptive tool switching to perception or action primitive tools.

Long-Horizon Open-vocabulary Manipulation Benchmarks. Manipulation benchmarks span tabletop manipulation (James et al., 2020; Liu et al., 2023; Tao et al., 2024; Zhu et al., 2020), household and kitchen environments (Li et al., 2022; Nasiriany et al., 2024), and language-conditioned long-horizon tasks (Han et al., 2025; Mees et al., 2022; Zhang et al., 2024), with real-to-sim suites measuring policy transfer (Kim et al., 2026; Li et al., 2024). While RoboCerebra (Han et al., 2025) and VLABench (Zhang et al., 2024) stress multi-step reasoning, they are low-fidelity, evaluate subtasks against a reset scene, and do not measure memory carried across them; closer to our memory axis, RMBench (Chen et al., 2026) targets memory-dependent manipulation but scopes it to short single-task contexts. RoboVoLo, built on RoboLab (Yang et al., 2026), instead requires reasoning over spatial state accumulated by earlier subtasks, isolating persistent memory as a measurable axis.

3. RoboVoLo Benchmark

Tasks and Scenes. Long-horizon, open-vocabulary manipulation requires a robot to reason and act over many steps. It must ground intent in scene context, track state as the scene changes, resolve fine-grained references, and apply world knowledge to carry out each step while monitoring and recovering from failures. This coupling of reasoning and execution is largely unsolved, and current benchmarks do not isolate it. RoboVoLo fills that gap with 126 tasks that span four reasoning categories, each requiring a chain of grounded manipulation actions. The tasks are built so they cannot be solved by obvious instruction-independent behavior. Figure 2 summarizes the taxonomy of four main categories:

1. **Commonsense grounding.** Success depends on understanding the functional or contextual role of objects in the current environment, rather than following the instruction verbatim.

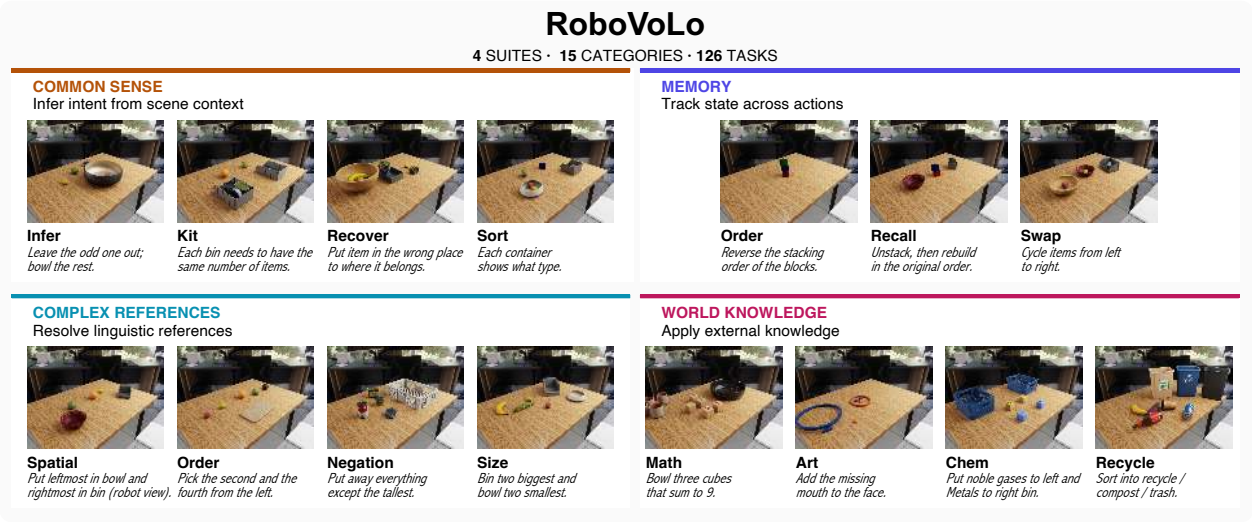

Figure 2: **RoboVoLo benchmark.** 126 long-horizon manipulation tasks across 15 categories, grouped into four capability suites: *Common Sense* (infer intent from scene context), *Memory* (track state across actions), *Complex References* (resolve spatial, ordinal, size, and negation cues), and *World Knowledge* (apply external knowledge spanning math, art, chemistry, and recycling). Each panel shows one representative task with its instruction.

2. **Memory.** These tasks require the policy to maintain information about earlier scene states during execution. Examples include restoring a previous arrangement, undoing a change, swapping objects, or rearranging objects relative to their initial configuration.
3. **Complex references.** Evaluate fine-grained language understanding. Instructions contain spatial, ordinal, relational, size-based, or negative references that disambiguate objects.
4. **World knowledge.** These tasks require general knowledge beyond the immediate geometry of the scene, covering domains like recycling, arithmetic, chemistry, and visual art.

Simulator. RoboVoLo is built on RoboLab (Yang et al., 2026), a high-fidelity simulation environment based on NVIDIA Isaac Lab (Mittal et al., 2025). To support these tasks, we expand RoboLab's asset library with 501 new objects: 247 household assets from NVIDIA's Lightwheel SimReady collection and 254 task-specific assets, including 118 chemical periodic-table element cubes, 120 geometric art objects varying in color, shape, and size, and 16 wooden math cubes with digits and operators. All assets include collision geometry and realistic physics materials, yielding a diverse collection spanning household, semantic, symbolic, and task-specific categories.

4. VoLoAgent and Physical Orchestration

4.1. Physical Orchestration

Virtual AI agents assume a world that holds still while the agent thinks, whereas a physical agent must reason while the world keeps moving. This imposes a core requirement: the agent must *monitor* the world for divergence between what it believes it has accomplished and the actual scene, *halt* an in-flight action as quickly as possible if divergence is detected, and *redirect* by choosing a correction: replanning, reissuing the action, or switching tools. Safe halting during reasoning may require an idling policy that for a fixed-base arm is simply stopping, but in general must keep the agent out of harm's way. We refer to this monitor–halt–redirect requirement as *physical orchestration*.

Prior closed-loop systems address parts of it: VLM-driven frameworks perform situated reasoning and failure recovery (Zhi et al., 2025), key-frame agents recover from execution errors (Nazarczuk et al., 2025), and reactive controllers halt a moving base to recover mid-task (Burgess-Limerick et al., 2023), but each targets a subset of

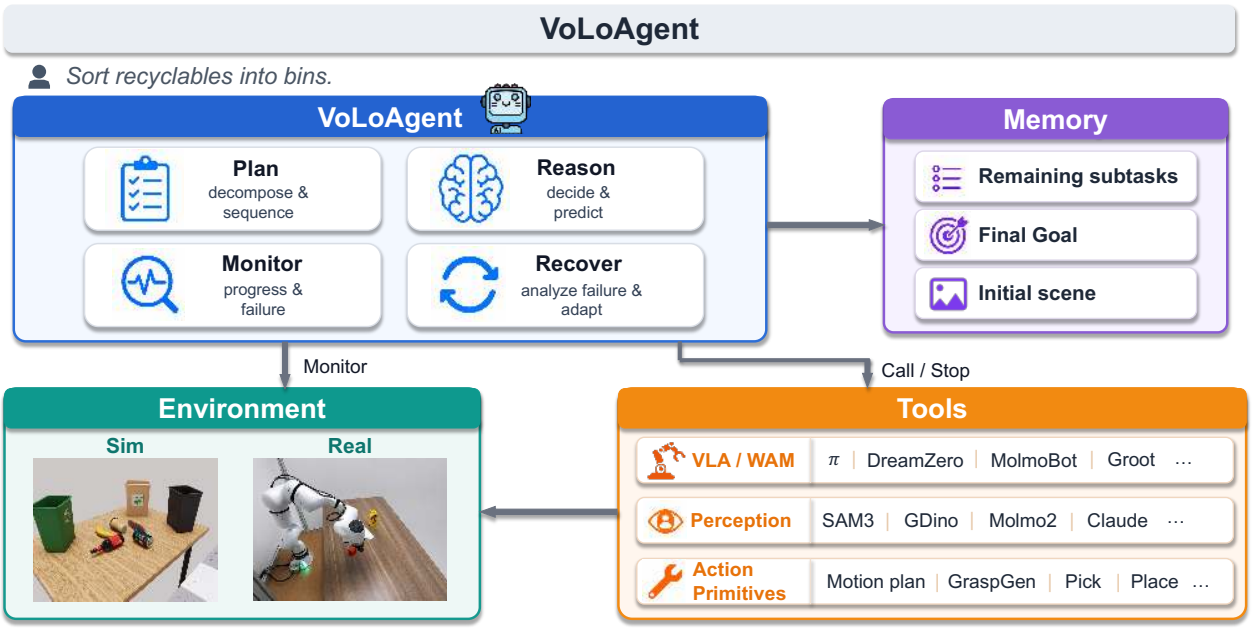

Figure 3: **VoLoAgent system.** A VLM agent plans, monitors, and orchestrates tools (VLA/WAM rollouts, perception models, grasp/place primitives) through one closed-loop control law. The agent can interrupt a VLA rollout and switch to a different tool when execution drifts.

these capabilities or a fixed pipeline. With *physical orchestration* we emphasize the need to handle all three together, for an open-vocabulary agent that switches tools mid-rollout, including interrupting asynchronous tools such as a learned visuomotor policy mid-rollout.

4.2. VoLoAgent System

VoLoAgent is a physical orchestrator: a single VLM agent that plans subtasks, monitors execution, and continuously routes among tools, deciding whether to continue, switch tools, advance, or recover. Unlike prior hierarchical systems that split control between a VLM planner and a VLA executor, here the VLA is one callable tool alongside perception models and grasp/place primitives, combined complementarily. It realizes the monitor–halt–recover loop through three design choices. **(P1) Asynchronous tools**: robot motion runs independent of the agent's reasoning, so the agent interleaves monitoring with execution rather than blocking. **(P2) Fast and slow memory**: a short monitor context (current observation, active subgoal, recent decisions) read as close to the motion timescale as possible (0.2Hz here), and a fuller deliberation context (task memory, scene history, tool catalog) consulted only at planning points, echoing dual-system VLA designs (Intelligence et al., 2025). **(P3) Safety-aware idling**: holding the robot still when reasoning must continue mid-task.

We instantiate three complementary tool families: **VLA/WAM** (e.g. $\pi_{0.5}$, DreamZero) is a first-class visuomotor tool but can struggle with open-vocabulary grounding. **Perception tools** (GroundingDINO (Liu et al., 2024), SAM2 (Ravi et al., 2024), SAM3 (Carion et al., 2025), Molmo2 (Clark et al., 2026; Deitke et al., 2024)) provide open-vocabulary detection and segmentation. **Action primitives** such as `grasp(target)` and `place(destination)` combine perception, GraspGen (Murali et al., 2025), and IK for geometry-grounded motion but remain rigid under contact-rich interaction. Full API signatures and prompts are in Appendices B and C. The VLM routes among these tools through the following phases:

Initial execution. Given a user instruction and the initial scene, the agent decomposes the task into atomic subgoals and stores them with the final goal and initial scene in external memory. It then issues the first tool call, typically a VLA rollout for its continuous visuomotor control, and begins monitoring concurrently.

Table 1: Results of various methods on our benchmark (rows: *Common Sense*, *Memory*, *Complex References*, *World Knowledge*), as well as on the *Robolab-Vague* benchmark. Methods (columns) are grouped by families: *Single action model* (no orchestrator), *Code-as-policy + VLM*, *TAMP + VLM*, and *VoLoAgent*. Each task is run for 3 episodes. All values are success rate (%, higher is better). **Bold** = best in row; underline = second-best.

		Single action model					*Code-as-policy*		*TAMP*	**VoLoAgent (Ours)**		
Suite	Category	$\pi_{0.5}$	π_0-FAST	MolmoBot	MolmoAct2	DreamZero	CaPX-s	CaPX-e	TiPToP	No VLA	Only VLA	**Full**
Common Sense	Infer	0.00	9.52	14.29	0.00	19.05	9.52	14.29	4.76	19.05	**52.38**	**52.38**
	Kit	16.67	4.17	0.00	0.00	12.50	12.50	16.67	8.33	41.67	33.33	**50.00**
	Recover	4.17	0.00	12.50	12.50	20.83	37.50	29.17	0.00	**62.50**	45.83	**62.50**
	Sort	23.81	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	47.62	**52.38**
	Overall	11.11	3.33	6.67	3.33	13.33	15.56	15.56	3.33	32.22	44.44	**54.44**
Memory	Order	12.50	25.00	33.33	25.00	29.17	16.67	16.67	0.00	25.00	29.17	**54.17**
	Recall	23.33	3.33	30.00	3.33	21.43	23.33	23.33	3.33	6.67	**63.33**	56.67
	Swap	3.33	0.00	6.67	3.33	0.00	6.67	6.67	0.00	**10.00**	**10.00**	3.33
	Overall	13.10	8.33	22.62	9.52	15.85	15.48	15.48	1.19	13.10	34.52	**36.90**
Complex References	Spatial	14.81	11.11	0.00	7.41	11.11	7.41	7.41	25.93	7.41	29.63	**40.74**
	Counting	16.67	12.50	12.50	0.00	0.00	4.17	4.17	12.50	4.17	45.83	**54.17**
	Negation	16.67	0.00	0.00	0.00	0.00	0.00	0.00	20.83	25.00	45.83	**54.17**
	Size+Sort	19.05	4.76	9.52	0.00	4.76	19.05	19.05	23.81	0.00	42.86	**57.14**
	Overall	16.67	7.29	5.21	2.08	4.17	7.29	7.29	20.83	9.38	40.62	**51.04**
World Knowledge	Art	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**16.67**	**16.67**	4.17	8.33
	Chem	8.33	0.00	12.50	4.17	12.50	4.17	4.17	50.00	29.17	41.67	**54.17**
	Math	4.17	0.00	0.00	0.00	0.00	0.00	0.00	4.17	**20.83**	0.00	12.50
	Recycle	25.00	0.00	4.17	0.00	0.00	4.17	4.17	20.83	0.00	**37.50**	25.00
	Overall	9.38	0.00	4.17	1.04	3.12	2.08	2.08	22.92	16.67	20.83	**25.00**
Robolab-Vague	Easy	19.79	10.94	13.76	6.25	19.79	16.67	15.10	29.69	19.79	**35.94**	34.90
	Med	17.54	11.40	11.40	6.14	18.80	14.04	9.65	7.02	16.67	26.32	**30.70**
	Hard	5.56	3.70	3.77	0.00	13.73	7.41	1.85	5.56	12.96	16.67	**24.07**
	Overall	16.94	10.00	11.52	5.28	18.61	14.44	11.39	18.89	17.78	30.00	**31.94**

Method abbreviations: CaPX-s = CaP-X single, CAPX-e = CAP-X ensemble.

Monitoring & routing. At each monitor step the agent reads the latest observation with memory under the monitor context (P2) and selects one of {CONTINUE, NEXT_SUBGOAL, RECOVERY}. There is no fixed split between planner and executor; the same agent decides whether to keep the current tool running, advance, or pause for recovery.

Recovery. On RECOVERY the active tool is idled (P3) and the agent enters the deliberation context to pick one of: CONTINUE if the alarm was a false positive (resume the rollout), REPLAN to re-issue the remaining subgoal decomposition, REWRITE to run the VLA with a new subgoal instruction, or GRASP / PLACE to run the respective primitive on a perception-grounded target.

The loop terminates on timeout or task completion. A key emergent property is complementarity: action primitives *inject* perception grounding into the VLA, so even a failed grasp leaves the gripper near the target with a clean view for the VLA to finish the pick (Sec. 5.3, 5.4).

5. Experimental Results

5.1. Setup

Simulation benchmarks. We evaluate on four RoboVoLo suites covering 126 tasks and on the existing RoboLab Yang et al. (2026) benchmark (120 tasks) with vague-choice instructions. All policy models use the DROID setup (Khazatsky et al., 2024): a 7-DoF Franka Research 3 arm with a Robotiq 2F-85 gripper, external ZED 2i and wrist ZED mini cameras, and a 7-DoF joint-position plus binary-gripper action space. Camera poses and lighting match the real DROID configuration. Each task is evaluated over three fixed-seed trials to ensure identical initial states across systems. We also explored MuJoCo-based benchmarks, including

LIBERO, RoboCerebra, and VLABench, but found them unsuitable due to limited generalist-policy support and insufficient realism; see Appendix H.

VoLoAgent. *VoLoAgent (Full)* uses Claude Opus 4.6 (Anthropic, 2026) as the decision-making VLM with the following tools: $\pi_{0.5}$ (Intelligence et al., 2025) as the VLA, SAM3 (Carion et al., 2025) and Molmo2 (Clark et al., 2026) as perception tools, and GraspGen (Murali et al., 2025) with multi-start IK plus depth-projected point placement for pick and place execution. The VLA and primitives run at 15Hz, while the VLM monitors at 0.2Hz from a front camera. We found this monitoring frequency reasonable for the pace of VLA motion, but increasing or adaptively varying it is an important future design goal. We compare two main ablations: *VoLoAgent (No VLA)*, which only uses perception tools and GRASP/PLACE action primitives, and *VoLoAgent (Only VLA)*, which disables all other tools and only relies on verbal steering of the VLA. Complete component ablations are in Sec. 5.4.

Baselines. We compare against three baseline families: (i) standalone action-model policies ($\pi_{0.5}$ (Intelligence et al., 2025), π_0-FAST (Pertsch et al., 2025), MolmoBot (Deshpande et al., 2026), MolmoAct2 (Fang et al., 2026), DreamZero (Ye et al., 2026)), (ii) code-as-policy + VLM (CaP-X (Fu et al., 2026), single and ensemble), and (iii) TAMP + VLM (TiPToP (Shen et al., 2026)).

5.2. Main Results

Table 1 shows that VoLoAgent achieves the best long-horizon open-vocabulary manipulation performance, outperforming single-model, code-as-policy, and TAMP baselines on every suite. The full system is significantly better than all methods ($p < 0.05$), except the Only VLA ablation ($p = 0.0598$), under a paired randomization test that asks whether one method consistently outperforms another across tasks (Edgington and Onghena, 2007) (see Appendix F). Against the strongest baseline in each suite, VoLoAgent (Full) gains +38.9% on Common Sense, +30.2% on Complex References, +14.3% on Memory, and +13.1% on Robolab-Vague; the exception is World Knowledge (+2.1%), where the TAMP baseline's symbolic planning is competitive. The gains come primarily from the planning, monitoring and recovery inherent in a physical orchestrator design, supplemented by the availability of complementary tools whose individual strengths cover others' blind spots. The full system also exceeds its own strongest ablation on every suite by between +1.9% and +10.4%.

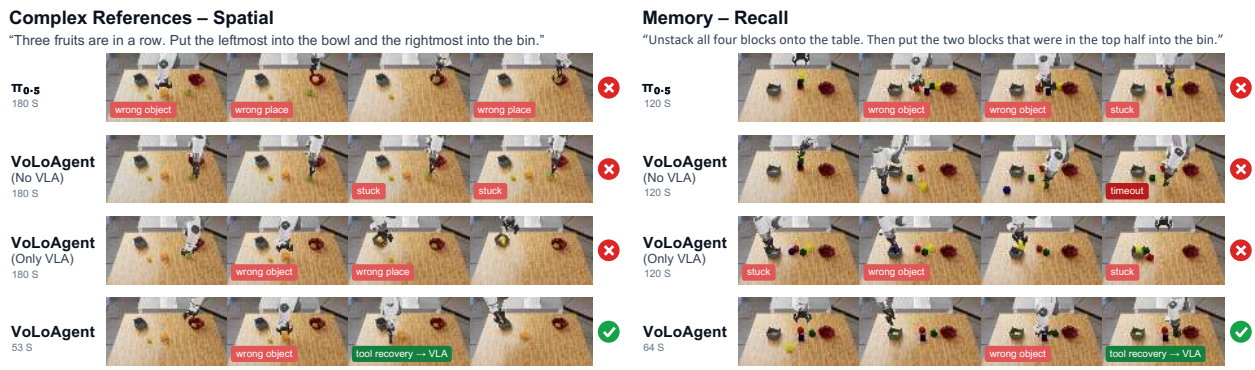

Figure 4: **Process comparison on two open-vocabulary long-horizon tasks**, one row per system. Red tags mark failure events and green tags mark grasp-tool recovery events. The behaviors shown are described in Sec. 5.2.

Figure 4 illustrates these gains on two representative tasks. $\pi_{0.5}$ relies on visual priors and ignores open-vocabulary constraints, placing all objects into the same bowl. VoLoAgent (No VLA) grounds the instruction and plans subtasks, but its action primitives struggle with contact-rich picks and exhaust the step budget. VoLoAgent (Only VLA) can steer the VLA through prompts, but remains limited by the VLA's perception errors, such as grasping an orange instead of a lemon. The full system combines their strengths: when the VLA

selects the wrong object, the grasp tool repositions the gripper on the correct target, and the VLA completes the contact-rich manipulation.

5.3. Failure Mode Analysis

Metrics Definition. We analyze failures along two axes. **World failures** measure state-level execution errors: wrong-object pick (*WOP*), wrong-target placement (*WTP*), and lack of end-effector progress for over 10s (*Stuck*), each paired with a `recovery` event when resolved. **VLM failures** measure reasoning and action errors: incorrect planning, false or missed completion monitor, missed failure detection, and wrong tool calling. Metrics mainly use ground-truth simulation states and human-labeled task features; full definitions are in Appendix K.

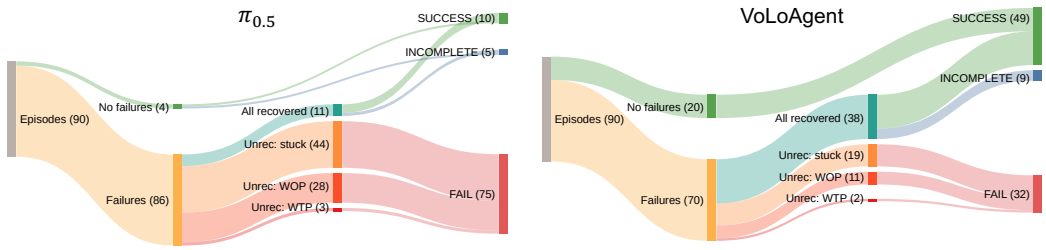

Figure 5: **World failure analysis** tracing episodes through failures, recovery, and outcomes for $\pi_{0.5}$ (left) and VoLoAgent (right). Major failure subtypes: *stuck*, *WOP*=wrong object picked, *WTP*=wrong target place. Band thickness is proportional to the number of episodes.

World Failures. Figure 5 traces $\pi_{0.5}$ and VoLoAgent through the failure-recovery pipeline for world failures. VoLoAgent has $5\times$ more failure-free episodes than $\pi_{0.5}$ (20 vs. 4). Among episodes that do hit a failure, VoLoAgent recovers from 54% ($38/70$) vs. only 13% ($11/86$) for $\pi_{0.5}$, showing that VoLoAgent not only enhances direct success but also greatly improves failure recovery (see Appendix J for VoLoAgent ablations).

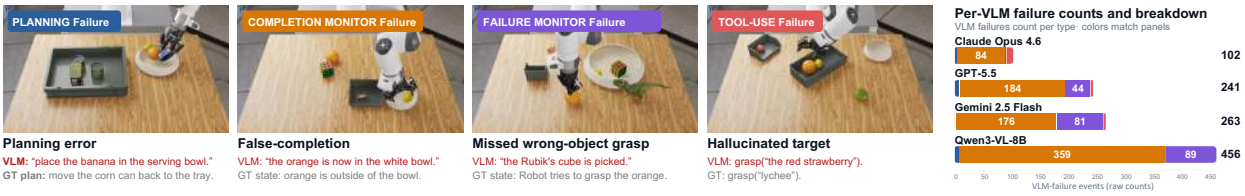

Figure 6: **VLM failure audit.** *Left:* one example per failure type (Planning, Completion-monitor, Failure-monitor, Tool-use). *Right:* per-VLM error counts across $n{=}90$ episodes; segment colors match the example tag colors. Qwen3-VL-8B reaches 23% of the ceiling error counts, Claude Opus 4.6 only 5%. Error definitions in Appendix K.

VLM Failures. Figure 6 shows one qualitative example per VLM failure class (left) and per-VLM event counts across four frontier VLMs (right). *Completion-monitor* errors dominate every backend, accounting for $>67\%$ of total events and increasing $4.3\times$ across VLM capability. *Failure-monitor* errors are another major class for every VLM except Claude Opus 4.6: GPT-5.5 has 44, Gemini 2.5 Flash 81, and Qwen3-VL-8B 89. *Planning* errors are rare for every backend (≤ 9 events per 90 episodes) as are *tool-use* mismatches (≤ 12 events). Improving completion and failure monitoring are next steps to strengthening the physical orchestrator design.

5.4. Component Ablations

We conduct comprehensive ablation studies, varying one component at a time while holding the rest at our default and study four axes: **System** comparing single VLA and VoLoAgent variants, **Perception** varying the

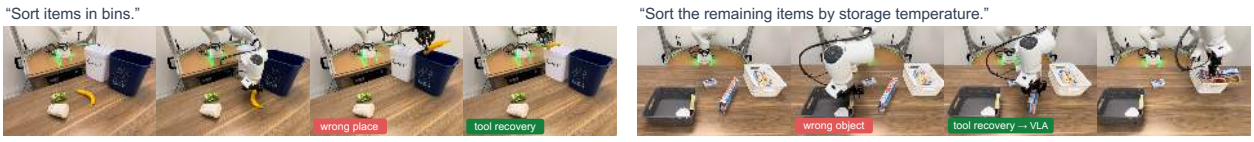

Figure 7: **Real robot examples.** VoLoAgent monitors and recovers from failures such as wrong place destination, wrong object pick in the real world as well.

perception tools and camera view, and the choice of **VLM** and **VLA** model. Table 2 reports cross-suite *Overall* success rates. **System.** The full system reaches 41.80, while Single-VLA, *VoLoAgent (No VLA)*, and *VoLoAgent (Only VLA)* score 12.57%, 17.76%, and 34.97% respectively. **Perception.** The system is robust to the choice of perception tool, with all variants achieving substantial gains. It also remains strong with the DROID exterior camera view to the orchestrator, though performance drops slightly because objects are sometimes occluded in the exterior and wrist views. **VLM.** Frontier VLMs as orchestrator yield +19% to +29% over the VLA-only baseline; with weaker VLMs the gain becomes marginal, the open-weights Qwen3-VL-8B model drops to +7%, aligning with its 4× higher VLM-failure count in Fig. 6. **VLA.** The orchestrator multiplies every VLA backbone by 2–6× overall, and gains hold across every base policy. We compared methods using task-paired success-rate differences, using aggregate success over all trials and two-sided exact sign-flip permutation tests over per-task success fractions. The only non-significant comparisons ($p < 0.05$) to the full system were two perceptual ablations: GDino+SAM2 / Molmo2 and Exterior camera.

Table 2: Component ablation, cross-suite *Overall* RoboVoLo success rate (%). Full breakdown in Table 4.

Axis	Ablation	**Success rate**
System	$\pi_{0.5}$ (Pure VLA)	12.57
	VoLoAgent (No VLA)	17.76
	VoLoAgent (Only VLA)	34.97
Perception	GDino+SAM2 / Molmo2	38.52
	SAM3 / VLM-point	36.07
	Exterior camera	36.94
VLM model	GPT-5.5	35.52
	Gemini-2.5-Flash	31.97
	Qwen3-VL-8B	19.95
VLA model	π_0-FAST	26.23
	MolmoBot-DROID	24.86
	DreamZero-DROID	21.86
VoLoAgent		**41.80**

5.5. Real Robot Validation

To evaluate whether VoLoAgent can operate beyond simulation, we deploy it on a real Franka FR3 with physical objects across a representative sample of 14 RoboVoLo tasks, running 3 matched-initial-state trials per task for $\pi_{0.5}$, VoLoAgent variants, and full VoLoAgent, for a total of 168 rollouts across variants. Full VoLoAgent achieves **42.9**% success versus 14.3% for $\pi_{0.5}$, a 3× improvement that supports the physical applicability of our agent loop design (Table 3). Figure 7 shows representative real-world recoveries from wrong-object picks and wrong-place drops. The intermediate variants achieve similar real-robot success (45.2% and 40.5%) with highly overlapping confidence intervals. Qualitatively, the grasp tool appears to work better in the real world than sim due to contact dynamics differences. Reaching statistical power to compare the ablations requires a larger real-robot study on substantially more tasks and trials per system. See Appendix G for more details including full list of tasks.

Table 3: Real-robot success rate (%) with 95% Wilson confidence across 14 tasks × 3 trials.

System	**Overall**	**95% CI**
$\pi_{0.5}$	14.3%	[6.7, 27.8]
VoLoAgent (No VLA)	45.2%	[31.2, 60.1]
VoLoAgent (Only VLA)	40.5%	[27.0, 55.5]
VoLoAgent (full)	42.9%	[29.1, 57.8]

6. Conclusion and Limitations

We introduced VoLoAgent, a physical orchestrator that unifies VLA/WAM rollouts, perception models, and grasp/place primitives in a VLM-managed closed loop, and RoboVoLo, a 126-task benchmark for open-vocabulary long-horizon manipulation. VoLoAgent outperforms existing baselines, with ablations showing that orchestration drives the gains.

Limitations. Our failure analysis (Sec. 5.3) highlights completion monitoring accuracy as a key direction for improvement. The per-call latency (∼1–5 s for cloud VLMs) of the orchestrating VLM bounds reaction time and may miss fast failures, calling for fast local monitors. VoLoAgent was demonstrated on a single-arm manipulator with a parallel-jaw gripper. Extending to bimanual, dexterous-hand, or mobile embodiments is supported by the framework, but requires retraining or swapping the VLA. Safe idling currently reduces to halting the arm, which does not generalize to embodiments that must act to stay safe (e.g., a balancing humanoid).

References

[1] Anthropic. Claude Opus 4.7 system card. https://www.anthropic.com/system-cards, 2026. Anthropic technical report. Also covers Claude Opus 4.6 and Claude Sonnet 4.6. 7

[2] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, Pete Florence, Chuyuan Fu, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alexander Herzog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Lisa Lee, Tsang-Wei Edward Lee, Sergey Levine, Yao Lu, Henryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael Ryoo, Grecia Salazar, Pannag Sanketi, Pierre Sermanet, Jaspiar Singh, Anikait Singh, Radu Soricut, Huong Tran, Vincent Vanhoucke, Quan Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Jialin Wu, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. RT-2: Vision-language-action models transfer web knowledge to robotic control. In *CoRL*, 2023. 3

[3] Ben Burgess-Limerick, Chris Lehnert, Jürgen Leitner, and Peter Corke. Enabling failure recovery for on-the-move mobile manipulation. In *IEEE ICRA Workshop on Robotic Perception and Mapping: Frontier Vision and Learning Techniques*, 2023. ICRA 2023 Workshop on Robot Failures; arXiv:2305.08351. 4

[4] Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, Rishi Hazra, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollár, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. SAM 3: Segment anything with concepts. *arXiv preprint arXiv:2511.16719*, 2025. 5, 7, 17

[5] Chi-Lam Cheang, Guangzeng Chen, Ya Jing, Tao Kong, Hang Li, Yifeng Li, Yuxiao Liu, Hongtao Wu, Jiafeng Xu, Yichu Yang, Hanbo Zhang, and Minzhao Zhu. GR-2: A generative video-language-action model with web-scale knowledge for robot manipulation. *arXiv preprint arXiv:2410.06158*, 2024. 2, 3

[6] Lingling Chen, Zongyao Lyu, and William J. Beksi. Reconvla: An uncertainty-guided and failure-aware vision-language-action framework for robotic control. *arXiv preprint arXiv:2604.16677*, 2026. 3

[7] Siyi Chen, Mikaela Angelina Uy, Chan Hee Song, Faisal Ladhak, Adithyavairavan Murali, Qing Qu, Stan Birchfield, Valts Blukis, and Jonathan Tremblay. SpaceTools: Tool-augmented spatial reasoning via double interactive rl. *CVPR*, 2026. 2

[8] Tianxing Chen, Yuran Wang, Mingleyang Li, Yan Qin, Hao Shi, Zixuan Li, Yifan Hu, Yingsheng Zhang, Kaixuan Wang, Yue Chen, Hongcheng Wang, Renjing Xu, Ruihai Wu, Yao Mu, Yaodong Yang, Hao Dong, and Ping Luo. RMBench: Memory-dependent robotic manipulation benchmark with insights into policy design. *arXiv preprint arXiv:2603.01229*, 2026. 3

[9] Christopher Clark, Jieyu Zhang, Zixian Ma, Jae Sung Park, Mohammadreza Salehi, Rohun Tripathi, Sangho Lee, Zhongzheng Ren, Chris Dongjoo Kim, Yinuo Yang, Vincent Shao, Yue Yang, Weikai Huang,

Ziqi Gao, Taira Anderson, Jianrui Zhang, Jitesh Jain, George Stoica, Winson Han, Ali Farhadi, and Ranjay Krishna. Molmo2: Open weights and data for vision-language models with video understanding and grounding. *arXiv preprint arXiv:2601.10611*, 2026. 5, 7, 17

[10] Yinpei Dai, Jayjun Lee, Nima Fazeli, and Joyce Chai. RACER: Rich language-guided failure recovery policies for imitation learning. *arXiv preprint arXiv:2409.14674*, 2024. 3

[11] Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, Jiasen Lu, Taira Anderson, Erin Bransom, Kiana Ehsani, Huong Ngo, YenSung Chen, Ajay Patel, Mark Yatskar, Chris Callison-Burch, Andrew Head, Rose Hendrix, Favyen Bastani, Eli VanderBilt, Nathan Lambert, Yvonne Chou, Arnavi Chheda, Jenna Sparks, Sam Skjonsberg, Michael Schmitz, Aaron Sarnat, Byron Bischoff, Pete Walsh, Chris Newell, Piper Wolters, Tanmay Gupta, Kuo-Hao Zeng, Jon Borchardt, Dirk Groeneveld, Crystal Nam, Sophie Lebrecht, Caitlin Wittlif, Carissa Schoenick, Oscar Michel, Ranjay Krishna, Luca Weihs, Noah A. Smith, Hannaneh Hajishirzi, Ross Girshick, Ali Farhadi, and Aniruddha Kembhavi. Molmo and PixMo: Open weights and open data for state-of-the-art vision-language models. *arXiv preprint arXiv:2409.17146*, 2024. 5

[12] Abhay Deshpande, Maya Guru, Rose Hendrix, Snehal Jauhri, Ainaz Eftekhar, Rohun Tripathi, Max Argus, Jordi Salvador, Haoquan Fang, Matthew Wallingford, Wilbert Pumacay, Yejin Kim, Quinn Pfeifer, Ying-Chun Lee, Piper Wolters, Omar Rayyan, Mingtong Zhang, Jiafei Duan, Karen Farley, Winson Han, Eli VanderBilt, Dieter Fox, Ali Farhadi, Georgia Chalvatzaki, Dhruv Shah, and Ranjay Krishna. MolmoBot: Large-scale simulation enables zero-shot manipulation. *arXiv preprint arXiv:2603.16861*, 2026. 3, 7

[13] Jiafei Duan, Wentao Yuan, Wilbert Pumacay, Yi Ru Wang, Kiana Ehsani, Dieter Fox, and Ranjay Krishna. Manipulate-anything: Automating real-world robots using vision-language models. In *CoRL*, 2024. 3

[14] Jiafei Duan, Wilbert Pumacay, Nishanth Kumar, Yi Ru Wang, Shulin Tian, Wentao Yuan, Ranjay Krishna, Dieter Fox, Ajay Mandlekar, and Yijie Guo. AHA: A vision-language-model for detecting and reasoning over failures in robotic manipulation. In *ICLR*, 2025. 3

[15] Eugene S. Edgington and Patrick Onghena. *Randomization Tests*. Chapman and Hall/CRC, Boca Raton, FL, 4 edition, 2007. 7

[16] Haoquan Fang, Jiafei Duan, Donovan Clay, Sam Wang, Shuo Liu, Weikai Huang, Xiang Fan, Wei-Chuan Tsai, Shirui Chen, Yi Ru Wang, Shanli Xing, Jaemin Cho, Jae Sung Park, Ainaz Eftekhar, Peter Sushko, Karen Farley, Angad Wadhwa, Cole Harrison, Winson Han, Ying-Chun Lee, Eli VanderBilt, Rose Hendrix, Suveen Ellawela, Lucas Ngoo, Joyce Chai, Zhongzheng Ren, Ali Farhadi, Dieter Fox, and Ranjay Krishna. MolmoAct2: Action reasoning models for real-world deployment. *arXiv preprint arXiv:2605.02881*, 2026. 3, 7

[17] Yunhai Feng, Jiaming Han, Zhuoran Yang, Xiangyu Yue, Sergey Levine, and Jianlan Luo. Reflective planning: Vision-language models for multi-stage long-horizon robotic manipulation. In *Conference on Robot Learning (CoRL)*, 2025. arXiv:2502.16707. 3

[18] Jiahui Fu, Junyu Nan, Lingfeng Sun, Hongyu Li, Jianing Qian, Jennifer L. Barry, Kris Kitani, and George Konidaris. NovaPlan: Zero-shot long-horizon manipulation via closed-loop video language planning. *arXiv preprint arXiv:2602.20119*, 2026. 3

[19] Max Fu, Justin Yu, Karim El-Refai, Ethan Kou, Haoru Xue, Huang Huang, Wenli Xiao, Guanzhi Wang, Fei-Fei Li, Guanya Shi, et al. CaP-X: A framework for benchmarking and improving coding agents for robot manipulation. *arXiv preprint arXiv:2603.22435*, 2026. 2, 3, 7

[20] Shenyuan Gao, William Liang, Kaiyuan Zheng, Ayaan Malik, Seonghyeon Ye, Sihyun Yu, Wei-Cheng Tseng, Yuzhu Dong, Kaichun Mo, Chen-Hsuan Lin, Qianli Ma, Seungjun Nah, Loic Magne, Jiannan Xiang,

Yuqi Xie, Ruijie Zheng, Dantong Niu, You Liang Tan, K. R. Zentner, George Kurian, Suneel Indupuru, Pooya Jannaty, Jinwei Gu, Jun Zhang, Jitendra Malik, Pieter Abbeel, Ming-Yu Liu, Yuke Zhu, Joel Jang, and Linxi Fan. DreamDojo: A generalist robot world model from large-scale human videos. *arXiv preprint arXiv:2602.06949*, 2026. 2, 3

[21] Qiao Gu, Yuanliang Ju, Shengxiang Sun, Igor Gilitschenski, Haruki Nishimura, Masha Itkina, and Florian Shkurti. SAFE: Multitask failure detection for vision-language-action models. *arXiv preprint arXiv:2506.09937*, 2025. 3

[22] Songhao Han, Boxiang Qiu, Yue Liao, Siyuan Huang, Chen Gao, Shuicheng Yan, and Si Liu. RoboCerebra: A large-scale benchmark for long-horizon robotic manipulation evaluation. In *NeurIPS*, 2025. 3

[23] Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, Pierre Sermanet, Noah Brown, Tomas Jackson, Linda Luu, Sergey Levine, Karol Hausman, and Brian Ichter. Inner monologue: Embodied reasoning through planning with language models. In *CoRL*, 2022. 3

[24] Physical Intelligence, Ali Amin, Raichelle Aniceto, Ashwin Balakrishna, Kevin Black, Ken Conley, Grace Connors, James Darpinian, Karan Dhabalia, Jared DiCarlo, et al. $\pi^*_{0.6}$: a VLA that learns from experience. *arXiv preprint arXiv:2511.14759*, 2025. 2, 3

[25] 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*, 2025. 2, 3, 5, 7

[26] Physical Intelligence, Bo Ai, Ali Amin, Raichelle Aniceto, Ashwin Balakrishna, Greg Balke, Kevin Black, George Bokinsky, Shihao Cao, Thomas Charbonnier, et al. $\pi_{0.7}$: a steerable generalist robotic foundation model with emergent capabilities. *arXiv preprint arXiv:2604.15483*, 2026. 2, 3

[27] Stephen James, Zicong Ma, David Rovick Arrojo, and Andrew J. Davison. RLBench: The robot learning benchmark & learning environment. *IEEE Robotics and Automation Letters*, 5(2):3019–3026, 2020. 3

[28] Yunfan Jiang, Agrim Gupta, Zichen Zhang, Guanzhi Wang, Yongqiang Dou, Yanjun Chen, Li Fei-Fei, Anima Anandkumar, Yuke Zhu, and Linxi Fan. VIMA: General robot manipulation with multimodal prompts. In *ICML*, 2023. 3

[29] Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, et al. DROID: A large-scale in-the-wild robot manipulation dataset. In *Robotics: Science and Systems (RSS)*, 2024. 6

[30] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024. 3

[31] Moo Jin Kim, Yihuai Gao, Tsung-Yi Lin, Yen-Chen Lin, Yunhao Ge, Grace Lam, Percy Liang, Shuran Song, Ming-Yu Liu, Chelsea Finn, and Jinwei Gu. Cosmos Policy: Fine-tuning video models for visuomotor control and planning. *arXiv preprint arXiv:2601.16163*, 2026. 2, 3

[32] Yejin Kim, Wilbert Pumacay, Omar Rayyan, Max Argus, Winson Han, Eli VanderBilt, Jordi Salvador, Abhay Deshpande, Rose Hendrix, Snehal Jauhri, Shuo Liu, Nur Muhammad Mahi Shafiullah, Maya Guru, Arjun Guru, Ainaz Eftekhar, Karen Farley, Donovan Clay, Jiafei Duan, Piper Wolters, Alvaro Herrasti, Ying-Chun Lee, Georgia Chalvatzaki, Yuchen Cui, Ali Farhadi, Dieter Fox, and Ranjay Krishna. MolmoSpaces: A large-scale open ecosystem for robot navigation and manipulation, 2026. URL https://arxiv.org/abs/2602.11337. 2, 3

[33] Jason Lee, Jiafei Duan, Haoquan Fang, Yuquan Deng, Shuo Liu, Boyang Li, Bohan Fang, Jieyu Zhang, Yi Ru Wang, Sangho Lee, Winson Han, Wilbert Pumacay, Angelica Wu, Rose Hendrix, Karen Farley, Eli VanderBilt, Ali Farhadi, Dieter Fox, and Ranjay Krishna. MolmoAct: Action reasoning models that can reason in space. *arXiv preprint arXiv:2508.07917*, 2025. 3

[34] Zixing Lei, Changxing Liu, Yichen Xiong, Minhao Xiong, Yuanzhuo Ding, Zhipeng Zhang, Weixin Li, and Siheng Chen. Towards Long-horizon Embodied Agents with Tool-Aligned Vision-Language-Action Models. *arXiv preprint arXiv:2605.13119*, 2026. 3

[35] Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Michael Lingelbach, Jiankai Sun, et al. BEHAVIOR-1K: A benchmark for embodied AI with 1,000 everyday activities and realistic simulation. In *Conference on Robot Learning (CoRL)*, 2022. 3

[36] Hao Li, Shuai Yang, Yilun Chen, Xinyi Chen, Xiaoda Yang, Yang Tian, Hanqing Wang, Tai Wang, Feng Zhao, Dahua Lin, and Jiangmiao Pang. Towards efficient and robust manipulation via multi-frame vision-language-action modeling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026. Oral. arXiv:2506.19816. 3

[37] 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. 2, 3

[38] Xuanlin Li, Kyle Hsu, Jiayuan Gu, Karl Pertsch, Oier Mees, Homer Rich Walke, Chuyuan Fu, Ishikaa Lunawat, Isabel Sieh, Sean Kirmani, Sergey Levine, Jiajun Wu, Chelsea Finn, Hao Su, Quan Vuong, and Ted Xiao. Evaluating real-world robot manipulation policies in simulation. In *CoRL*, 2024. 3

[39] Yi Li, Yuquan Deng, Jesse Zhang, Joel Jang, Marius Memmel, Raymond Yu, Caelan Reed Garrett, Fabio Ramos, Dieter Fox, Anqi Li, Abhishek Gupta, and Ankit Goyal. HAMSTER: Hierarchical action models for open-world robot manipulation, 2025. 2, 3

[40] Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. Code as policies: Language model programs for embodied control. In *ICRA*, 2023. 2, 3

[41] Zijun Lin, Jiafei Duan, Haoquan Fang, Dieter Fox, Ranjay Krishna, Cheston Tan, and Bihan Wen. FailSafe: Reasoning and recovery from failures in vision-language-action models. *arXiv preprint arXiv:2510.01642*, 2025. 2, 3

[42] Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2023. 2, 3

[43] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. Grounding DINO: Marrying DINO with grounded pre-training for open-set object detection. In *ECCV*, 2024. 5, 17

[44] 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 *ICLR*, 2025. 3

[45] Zhen Liu, Xinyu Ning, Zhe Hu, Xinxin Xie, Weize Li, Zhipeng Tang, Chongyu Wang, Zejun Yang, Hanlin Wang, Yitong Liu, and Zhongzhu Pu. Goal2Skill: Long-horizon manipulation with adaptive planning and reflection. *arXiv preprint arXiv:2604.13942*, 2026. 2, 3

[46] Weijia Liufu, Xiaoyu Guo, Ruiyi Chen, Jingzhi Liu, Kaidong Zhang, Xiwen Liang, Jianqi Lin, Dawei Sun, Yuze Wang, Rongtao Xu, Bingqian Lin, Bowen Yang, Tongtong Cao, Bowen Peng, Dongyu Zhang, Guangrun Wang, Min Wang, Liang Lin, and Xiaodan Liang. Repo-vla: Recovery-driven policy optimization for vision-language-action models. *arXiv preprint arXiv:2605.09410*, 2026. 3

[47] Guoqing Ma, Siheng Wang, Zeyu Zhang, Shan Yu, and Hao Tang. Generalvla: Generalizable vision–language–action models with knowledge-guided trajectory planning. *arXiv preprint arXiv:2602.04315*, 2026. 3

[48] Oier Mees, Lukas Hermann, Erick Rosete-Beas, and Wolfram Burgard. CALVIN: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters*, 7(3):7327–7334, 2022. 3

[49] Aoran Mei, Guo-Niu Zhu, Huaxiang Zhang, and Zhongxue Gan. ReplanVLM: Replanning robotic tasks with visual language models. *arXiv preprint arXiv:2407.21762*, 2024. 3

[50] Mayank Mittal, Pascal Roth, James Tigue, Antoine Richard, Octi Zhang, Peter Du, Antonio Serrano-Muñoz, Xinjie Yao, René Zurbrügg, Nikita Rudin, et al. Isaac lab: A gpu-accelerated simulation framework for multi-modal robot learning. *arXiv preprint arXiv:2511.04831*, 2025. doi: 10.48550/arXiv.2511.04831. URL https://arxiv.org/abs/2511.04831. 4

[51] Adithyavairavan Murali, Balakumar Sundaralingam, Yu-Wei Chao, Wentao Yuan, Jun Yamada, Mark Carlson, Fabio Ramos, Stan Birchfield, Dieter Fox, and Clemens Eppner. GraspGen: A diffusion-based framework for 6-DoF grasping with on-generator training. *arXiv preprint arXiv:2507.13097*, 2025. 5, 7, 17

[52] Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. RoboCasa: Large-scale simulation of everyday tasks for generalist robots. In *Robotics: Science and Systems (RSS)*, 2024. 3

[53] Michal Nazarczuk, Jan Kristof Behrens, Karla Stepanova, Matej Hoffmann, and Krystian Mikolajczyk. Closed loop interactive embodied reasoning for robot manipulation. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2025. 4

[54] Svyatoslav Pchelintsev, Maxim Patratskiy, Anatoly Onishchenko, Alexandr Korchemnyi, Aleksandr Medvedev, Uliana Vinogradova, Ilya Galuzinsky, Aleksey Postnikov, Alexey K. Kovalev, and Aleksandr I. Panov. LERa: Replanning with visual feedback in instruction following. *arXiv preprint arXiv:2507.05135*, 2025. 3

[55] Karl Pertsch, Kyle Stachowicz, Brian Ichter, Danny Driess, Suraj Nair, Quan Vuong, Oier Mees, Chelsea Finn, and Sergey Levine. FAST: Efficient action tokenization for vision-language-action models. *arXiv preprint arXiv:2501.09747*, 2025. 7

[56] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. SAM 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 5

[57] André Schakkal, Ben Zandonati, Zhutian Yang, and Navid Azizan. Hierarchical vision-language planning for multi-step humanoid manipulation. In *Robotics: Science and Systems (RSS) Workshop on Robot Planning in the Era of Foundation Models*, 2025. arXiv:2506.22827. 3

[58] William Shen, Nishanth Kumar, Sahit Chintalapudi, Jie Wang, Christopher Watson, Edward Hu, Jing Cao, Dinesh Jayaraman, Leslie Pack Kaelbling, and Tomás Lozano-Pérez. TiPToP: A modular open-vocabulary planning system for robotic manipulation. *arXiv preprint arXiv:2603.09971*, 2026. 3, 7

[59] Hao Shi, Bin Xie, Yingfei Liu, Lin Sun, Fengrong Liu, Tiancai Wang, Erjin Zhou, Haoqiang Fan, Xiangyu Zhang, and Gao Huang. Memoryvla: Perceptual-cognitive memory in vision-language-action models for robotic manipulation. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2508.19236. 3

[60] Lucy Xiaoyang Shi, Brian Ichter, Michael Equi, Liyiming Ke, Karl Pertsch, Quan Vuong, James Tanner, Anna Walling, Haohuan Wang, Niccolo Fusai, Adrian Li-Bell, Danny Driess, Lachy Groom, Sergey Levine, and Chelsea Finn. Hi robot: Open-ended instruction following with hierarchical vision-language-action models. *arXiv preprint arXiv:2502.19417*, 2025. 2, 3

[61] Ishika Singh, Valts Blukis, Arsalan Mousavian, Ankit Goyal, Danfei Xu, Jonathan Tremblay, Dieter Fox, Jesse Thomason, and Animesh Garg. ProgPrompt: Generating situated robot task plans using large language models. In *ICRA*, 2023. 2, 3

[62] Marta Skreta, Zihan Zhou, Jia Lin Yuan, Kourosh Darvish, Alán Aspuru-Guzik, and Animesh Garg. RePLan: Robotic replanning with perception and language models. *arXiv preprint arXiv:2401.04157*, 2024. 3

[63] Stone Tao, Fanbo Xiang, Arth Shukla, Yuzhe Qin, Xander Hinrichsen, Xiaodi Yuan, Chen Bao, Xinsong Lin, Yulin Liu, Tse-kai Chan, Yuan Gao, Xuanlin Li, Tongzhou Mu, Nan Xiao, Arnav Gurha, Zhiao Huang, Roberto Calandra, Rui Chen, Shan Luo, and Hao Su. ManiSkill3: GPU parallelized robotics simulation and rendering for generalizable embodied AI. *arXiv preprint arXiv:2410.00425*, 2024. 3

[64] Tianshuo Yang, Guanyu Chen, Yutian Chen, Zhixuan Liang, Yitian Liu, Zanxin Chen, Chunpu Xu, Haotian Liang, Jiangmiao Pang, Yao Mu, and Ping Luo. Hivla: A visual-grounded-centric hierarchical embodied manipulation system. *arXiv preprint arXiv:2604.14125*, 2026. 3

[65] Xuning Yang, Rishit Dagli, Alex Zook, Hugo Hadfield, Ankit Goyal, Stan Birchfield, Fabio Ramos, and Jonathan Tremblay. RoboLab: A high-fidelity simulation benchmark for analysis of task generalist policies. *RSS*, 2026. 2, 3, 4, 6, 18

[66] Yifan Yang, Zhixiang Duan, Tianshi Xie, Fuyu Cao, Pinxi Shen, Peili Song, Chenyang Zhao, Piaopiao Jin, Guokang Sun, Shaoqing Xu, Yangwei You, and Jingtai Liu. Fpc-vla: A vision-language-action framework with a supervisor for failure prediction and correction. *Expert Systems with Applications*, 316:131742, 2026. arXiv:2509.04018. 3

[67] Zhejian Yang, Yongchao Chen, Xueyang Zhou, Jiangyue Yan, Dingjie Song, Yinuo Liu, Yuting Li, Yu Zhang, Pan Zhou, Hechang Chen, and Lichao Sun. Agentic robot: A brain-inspired framework for vision-language-action models in embodied agents. *arXiv preprint arXiv:2505.23450*, 2025. 2, 3

[68] Zhutian Yang, Caelan Garrett, Dieter Fox, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Guiding long-horizon task and motion planning with vision language models. *arXiv preprint arXiv:2410.02193*, 2024. 3

[69] Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, Ayaan Malik, Kyungmin Lee, William Liang, Nadun Ranawaka, Jiasheng Gu, Yinzhen Xu, Guanzhi Wang, Fengyuan Hu, Avnish Narayan, Johan Bjorck, Jing Wang, Gwanghyun Kim, Dantong Niu, Ruijie Zheng, Yuqi Xie, Jimmy Wu, Qi Wang, Ryan Julian, Danfei Xu, Yilun Du, Yevgen Chebotar, Scott Reed, Jan Kautz, Yuke Zhu, Linxi Fan, and Joel Jang. World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*, 2026. 2, 3, 7

[70] Zewei Ye, Weifeng Lu, Minghao Ye, Tao Lin, Shuo Yang, Junchi Yan, and Bo Zhao. RoboFAC: A comprehensive framework for robotic failure analysis and correction. *arXiv preprint arXiv:2505.12224*, 2025. 3

[71] Pengfei Yi, Yingjie Ma, Wenjiang Xu, Yanan Hao, Shuai Gan, Wanting Li, and Shanlin Zhong. Critic in the loop: A tri-system VLA framework for robust long-horizon manipulation. *arXiv preprint arXiv:2603.05185*, 2026. 2, 3

[72] Jianke Zhang, Yanjiang Guo, Xiaoyu Chen, Yen-Jen Wang, Yucheng Hu, Chengming Shi, and Jianyu Chen. HiRT: Enhancing robotic control with hierarchical robot transformers. In *CoRL*, 2024. 3

[73] 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. 3

[74] Peiyuan Zhi, Zhiyuan Zhang, Yu Zhao, Muzhi Han, Zeyu Zhang, Zhitian Li, Ziyuan Jiao, Baoxiong Jia, and Siyuan Huang. Closed-loop open-vocabulary mobile manipulation with GPT-4V. In *ICRA*, 2025. 3, 4

[75] Yuke Zhu, Josiah Wong, Ajay Mandlekar, Roberto Martín-Martín, Abhishek Joshi, Kevin Lin, Abhiram Maddukuri, Soroush Nasiriany, and Yifeng Zhu. robosuite: A modular simulation framework and benchmark for robot learning. *arXiv preprint arXiv:2009.12293*, 2020. 2, 3