InternVLA-M1: Latent Spatial Grounding for Instruction-Following Robotic Manipulation

Intern Robotics, Shanghai AI Laboratory

*We introduce InternVLA-M1, a unified framework for spatial grounding and robot control that advances instruction-following robots toward general-purpose intelligence. Its core idea is **spatially guided vision-language-action training**, where spatial grounding serves as the critical link between instructions and robot actions. InternVLA-M1 employs a two-stage pipeline: (i) spatial grounding pre-training on over 2.3M spatial reasoning data to determine "where to act" by aligning instructions with visual, embodiment-agnostic positions, and (ii) spatially guided action post-training to decide "how to act" by generating embodiment-aware actions through plug-and-play **spatial prompting**. This spatially guided training recepit yields consistent gains: InternVLA-M1 outperforms its variant without spatial guidance by +13.6% on SimplerEnv Google Robot, +17% on WidowX, and +4.3% on LIBERO Franka. To further scale instruction following, we built a simulation engine to collect 244K pick-and-place episodes, enabling a 6.2% average improvement across 200 tasks and 3K+ objects. In real-world clustered pick-and-place, InternVLA-M1 improved by 7.3%, and with synthetic co-training, achieved +20.6% on unseen objects and novel configurations. Moreover, in long-horizon reasoning-intensive scenarios, it surpassed existing works by over 10 points. These results highlight spatially guided training as a unifying principle for scalable and resilient generalist robots.*

Code:InternVLA-M1 | Model:InternVLA-M1 | Data:InternData-M1 | Homepage

1. Introduction

Large multimodal foundation models Bai et al. (2025b); Chen et al. (2024); Li et al. (2024b); Radford et al. (2021); Zhai et al. (2023) have demonstrated strong generalization by leveraging web-scale vision–language alignment and instruction-following corpora. To extend these capabilities into the physical domain, robots must not only understand *what* an instruction means but also determine *where* and *how* to act in the 3D world. This gap is fundamental. Textual abstractions capture spatial cues only indirectly, whereas real-world actions demand continuous, embodied interactions that are scarcely represented in the training data of vision–language models (VLMs). Teleoperated datasets Bu et al. (2025a); Collaboration et al. (2023); Khazatsky et al. (2024); Wu et al. (2024) provide valuable supervision; yet, their scale and diversity remain modest compared to large instruction-following corpora. In this context, an embodiment-agnostic spatial prior, which functions as a bridge between textual instructions and embodiment-specific motor commands, offers a promising foundation for scalable robot learning.

Prior work has approached this challenge through hierarchical robotic systems Cao et al. (2025); Huang et al. (2024a, 2023, 2024b); Liu et al. (2024); Qi et al. (2025); Yuan et al. (2024), which explicitly encode spatial priors using foundation models Fang et al. (2023); Kirillov et al. (2023); Oquab et al. (2023) but often rely on rule-based task decomposition and manually designed planning heuristics. This rigid separation between symbolic task structures and low-level motor control makes such systems difficult to scale automatically to more complex and diverse tasks, particularly hindering end-to-end policy learning. In contrast, recent data-driven VLAs AI (2024); Black et al. (2024); Brohan et al. (2023); Kim et al. (2024); Lee et al. (2025); Shi et al. (2025) leverage pretrained vision-language models and large-scale teleoperation datasets Bu et al. (2025a); Collaboration et al. (2023); Khazatsky et al. (2024); Wu et al. (2024) to directly learns robot control. However, these

Contributors are listed in Appendix A.

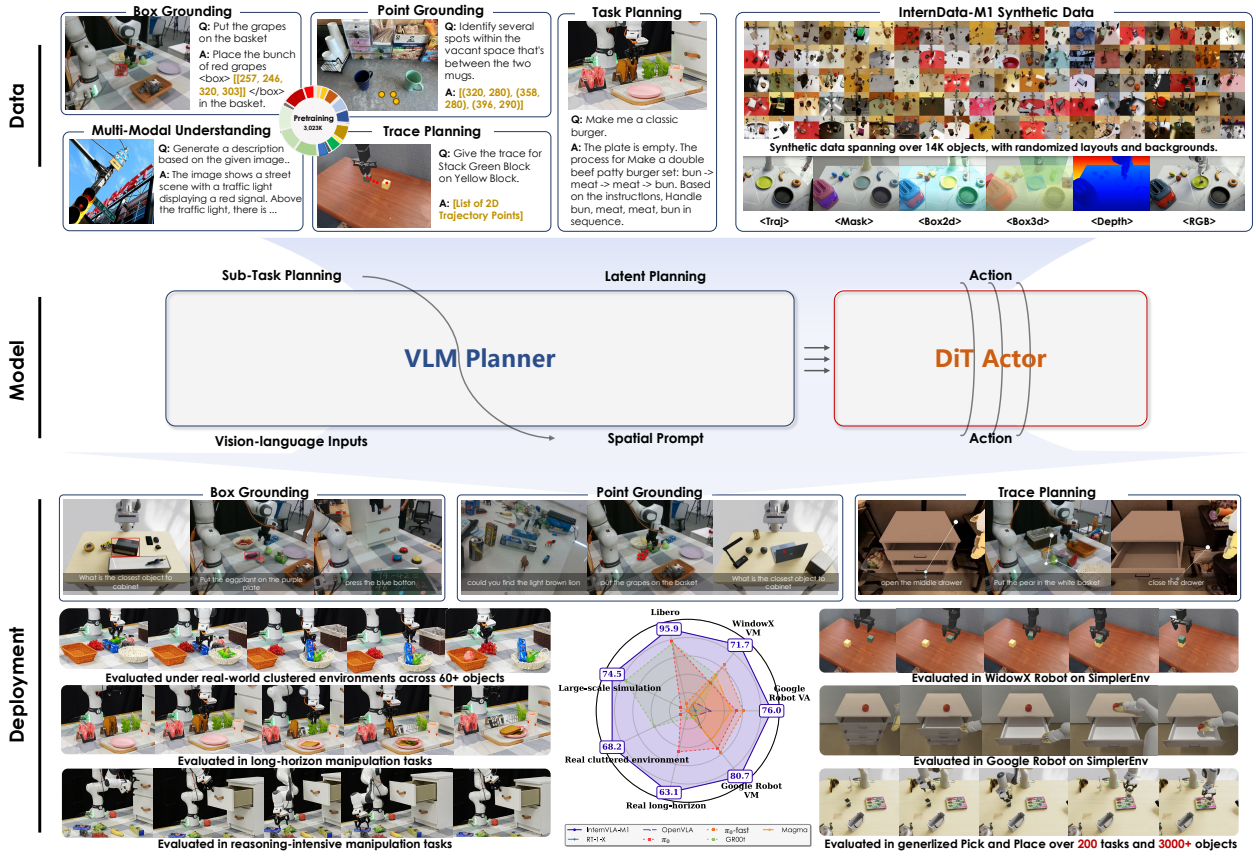


Figure 1. InternVLA-M1 integrates spatial grounding into the vision–language–action training pipeline. Given a task instruction, the VLM planner produces latent plans through explicit spatial prompting, which then effectively guides the action expert to generate control signals.

models tend to overfit fine-grained motor behaviors while under-generalizing to high-level linguistic instructions that involve absolute or relational positions, thereby failing to fully incorporate spatial priors into execution. Core spatial priors such as object recognition, affordance grounding, visual trajectory reasoning, relative localization, and scaling provide transferable knowledge across robotic platforms. Once these priors are established, embodiment-specific learning can focus on concrete control strategies (e.g., manipulator joints, end-effector trajectories, humanoid locomotion, or mobile navigation). Such a division clarifies the role of spatial priors as general-purpose foundations while leaving embodiment-specific details to downstream adaptation, thereby bridging the gap between abstract instruction following and grounded physical execution.

Building on the separation of spatial priors and embodiment-specific control, we propose InternVLA-M1, a dual-system framework built on top of a spatial prior VLM planner and an action expert for reliable instruction-following manipulation, as shown in Figure 1. InternVLA-M1 explicitly decomposes the problem into two stages: (i) spatial grounding pre-training for the VLM, which establishes transferable spatial priors through scalable point, box, and trace prediction; and (ii) spatially guided action post-training, which specializes these priors into embodiment-specific motor policies via spatial prompting as an action condition. Experimental results demonstrate that, without requiring paired spatial reasoning and action data, InternVLA-M1 enables efficient downstream post-training, improves generalization to unseen objects and paraphrased instructions, and delivers robust real-world performance in out-of-distribution environments.

To validate these capabilities, we design a comprehensive evaluation suite spanning multimodal understanding, robotic grounding, simulation benchmarks, and real-world tasks:

- On SimplerEnv (Google Robot, WidowX), InternVLA-M1 establishes new state of the art, improving average success by up to +5.9% and +9.8%, respectively.
- On the large-scale benchmark of 200 tabletop tasks, our model exhibits strong generalization to unseen objects and unseen instructions under few-demonstrations fine-tuning, achieving an average improvement of 6.2% over the previous state-of-the-art.
- In real-world settings, InternVLA-M1 attains 92% success in-distribution and sustains robust long-horizon performance under perturbations (e.g., physical interference, task replanning), outperforming baselines such as GR00T and π_0 by large margins.

2. InternVLA-M1

We propose InternVLA-M1, a dual-system, end-to-end vision–language–action (VLA) framework. It integrates both a language head and an action head within a single model (Section 2.1). The language head establishes instruction-to-visual grounding through spatial pretraining and co-training, while the action head conditions on these learned spatial priors to generate embodiment-specific motor commands(Section 2.2). This joint design bridges abstract linguistic goals with grounded execution, enabling robust instruction following across diverse and complex scenes.

2.1. Model Architecture

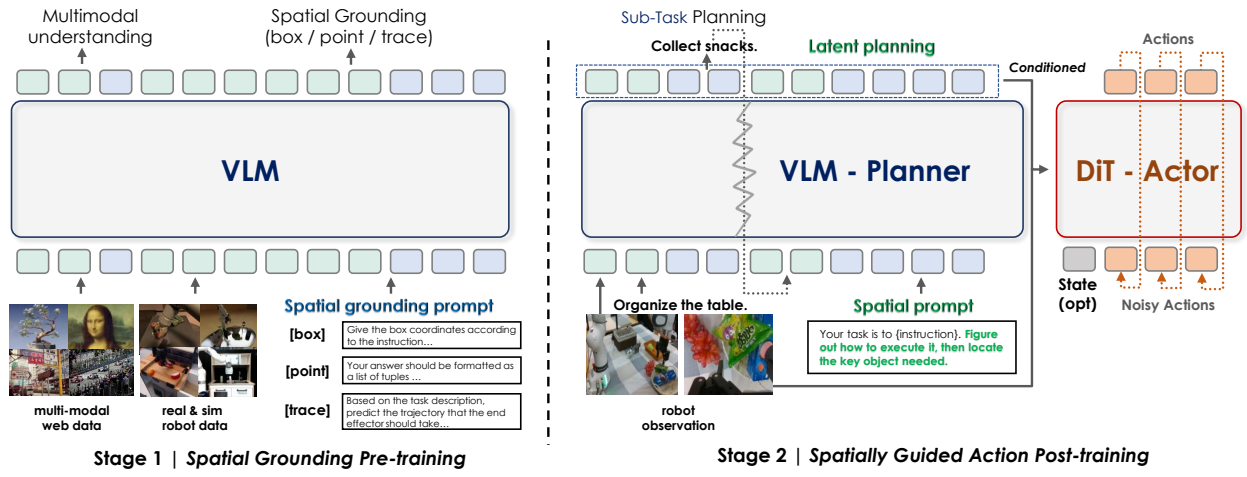


Figure 2. **Overview of InternVLA-M1.** InternVLA-M1 adopts a spatially guided two-stage training pipeline. Stage 1 (spatial grounding pre-training): the VLM is trained on large-scale multisource multimodal spatial grounding data to learn embodiment-agnostic spatial priors. Stage 2 (spatially guided action post-training): the VLM Planner, functioning as a slow but reliable System 2 reasoner, generates latent planning tokens via spatial prompting as the condition to the action expert (instantiated as a DiT Actor) to execute as a fast System 1 controller.

Dual-System. InternVLA-M1 is a dual-system, end-to-end VLA framework pre-trained on large-scale spatial grounding data collected from diverse sources. InternVLA-M1 employs the Qwen2.5-VL-3B-instruct Bai et al. (2025a) as the multimodal encoder for System 2, which is to capture spatial priors. It adopts the diffusion policy Chi et al. (2023) (86 MB) as the Action Expert (System 1, the fast executor), which effectively models embodiment-specific control. This expert is built on the DINOv2

visual encoder Oquab et al. (2023) (21 MB) and a lightweight state encoder (0.4 MB), forming a compact vision–action model. In total, InternVLA-M1 comprises approximately 4.1B parameters. During inference, the system runs on a single RTX 4090 GPU with around 12 GB of memory usage. With FlashAttention, the VLM component achieves inference speeds of approximately 10 FPS. Action execution can be further accelerated via chunking and KV caching.

Dual-Supervision. The dual-system architecture supports both multimodal supervision and action supervision during training. In each training step, batches from both data types are jointly processed, and the model computes losses from the two supervision signals. The resulting gradients are aggregated and applied in a single optimization update, ensuring that perception and control are co-adapted rather than learned in isolation. Specifically, the VLM planner is aligned with a broad range of spatial grounding data, both real and synthetic, covering tasks such as object detection, affordance recognition, and visual trajectory planning. In parallel, the Action Expert is trained on robot demonstration data, enabling it to specialize these priors into embodiment-specific motor commands. This dual-supervision strategy establishes a cohesive link between high-level semantic perception and low-level motion control, which is essential for robust instruction following in both simulation and real-world settings.

Latent Planning via Spatial Prompting. To connect the VLM Planner with the action expert, we adopt a lightweight querying transformer (8.7 MB) conditioned on the latent planning embeddings produced by the VLM Planner. The querying transformer stabilizes expert learning and inference by mapping variable-length input tokens into a fixed set of learnable query tokens. It is implemented as a k-layer cross-attention module, where the query tokens selectively attend to k intermediate layers of the VLM (e.g., $k = 1$ attends only to the final layer).

To explicitly activate the spatial perception capability learned during spatial grounding pre-training, we employ spatial prompting. For instance, in general object manipulation tasks, we append simple prompts such as "Figure out how to execute it, then locate the key object needed." after the task instruction. The extracted feature embeddings provide the planner with explicit spatial cues that facilitate more reliable grounding. Motivated by prior studies Bjorck et al. (2025); Driess et al. (2025); Zhou et al. (2025b) showing that direct gradient flow between action and VLM modules may distort multimodal knowledge, we introduce a gradient decay factor within the querying transformer. This attenuates the gradients propagated from the Action Expert back to the VLM (e.g., by a factor of 0.5), thereby preserving the Planner's semantic reasoning ability while still enabling effective joint optimization.

2.2. Training Recipe

To leverage spatial priors for stronger embodiment-specific control in instruction following, InternVLA-M1 adopts a spatially guided two-stage training pipeline:

Stage 1: Spatial Grounding Pre-training. As shown in Figure 2, the first stage optimizes only the VLM. The objective is not generic vision–language pre-training, but stronger spatial reasoning and planning ability essential for robotics. We combine internet-scale multimodal corpora with robot-specific datasets such as RefCOCO, RoboRefIt Lu et al. (2023), A0 Xu et al. (2025b), MolmoAct Lee et al. (2025), and Pixmo-Points Deitke et al. (2024). All robot datasets are reformatted into a unified QA-style structure covering bounding-box detection, trajectory prediction, affordance recognition, and chain-of-thought reasoning. Aligning them with web-scale data enables training under the same supervised fine-tuning framework as conventional VLMs.

Stage 2: Spatially Guided Action Post-training. In this stage, both the VLM and Action Expert are jointly optimized on demonstration data, ensuring semantic understanding and motion generation

remain tightly integrated. Two strategies are employed:

- **Spatial prompting.** Before predicting actions, we prepend a spatial cue to the task instruction to elicit structured reasoning about object relationships and task constraints. For example, the instruction "store all toys into the toy box" can be augmented with: "Identify all relevant toys and their spatial relationships to the container." Although the VLM does not explicitly output a response to this auxiliary cue, its inclusion improves spatial awareness and generalization in manipulation tasks.
- **Co-training with spatial grounding data.** Training alternates between robot trajectory data and grounding data. For trajectory data, both the VLM backbone and the action Expert are optimized with an L2 loss between predicted and ground-truth noise. For spatial grounding data, only the VLM backbone is updated via next-token prediction. This co-training scheme reinforces spatial reasoning while supporting efficient end-to-end optimization.

3. Data

This section introduces the datasets used in InternVLA-M1, covering pre-training, mid-training, and post-training stages. For VLM pre-training, we construct large-scale spatial grounding datasets with point, box, and trajectory annotations to enhance spatial perception and vision-language alignment. Mid-training employs synthetic manipulation data to bridge pre-training knowledge and robotic execution. Post-training uses both simulated and real-world instruction-following data, including large-scale tabletop tasks and real-robot demonstrations for long-horizon manipulation.

3.1. Spatial Grounding Data for Pre-training

The multimodal training dataset for our model comprises over 3M data, categorized into four distinct types: General Question Answering (General QA), Bounding Box Question Answering (Box QA), Trajectory Question Answering (Trajectory QA), and Point Question Answering (Point QA), as shown in Figure 3. Notably, more than 2.3M of these data are dedicated to spatial reasoning datasets. These categories ensure robust multimodal understanding while supporting adaptation to embodied tasks in tabletop robotic scenarios. Below, we describe each category:

- **General QA**. Sourced from LLaVA-OneVision Li et al. (2024a) and InternVL3 Chen et al. (2024); Zhu et al. (2025), this category is sampled to cover diverse multimodal tasks, including image captioning, visual question answering (VQA), optical character recognition (OCR), knowledge grounding, and creative writing.
- **Bounding Box QA**. We curate a diverse collection of multimodal grounding datasets, including RefCOCO Mao et al. (2016); Yu et al. (2016), ASv2 Wang et al. (2024), and COCO-ReM Singh et al. (2024), sourced from InternVL3 Chen et al. (2024); Zhu et al. (2025). Additionally, we incorporate the InternData-M1 dataset, generated via scalable synthetic data generation as Sec. 3.3, and the RoboRefIt dataset Lu et al. (2023), a specialized dataset for robotics grounding.
- **Trajectory QA**. This category integrates the A0 ManiSkill subset Xu et al. (2025a), the InternData-M1 trajectory point dataset, and the MolmoAct dataset Lee et al. (2025) to enable precise end-effector trajectory prediction. The A0 ManiSkill subset provides high-quality, object-centric trajectory data, where small objects move in coordination with the robotic arm's gripper. These trajectories can be approximated as end-effector movements for tabletop manipulation tasks.
- **Point QA**. For precise point localization, we integrate multiple datasets, including the Pixmo-Points dataset Deitke et al. (2024), the RoboPoint dataset Yuan et al. (2024), the RefSpatial dataset Zhou et al. (2025a), and a point subset extracted from the InternData-M1 dataset, each subjected to tailored preprocessing. Specifically, the Pixmo-Points dataset is filtered to exclude images with

resolutions exceeding 1024 pixels and restricted to a maximum of 10 points per image. Additionally, we prioritize the extraction of object reference and region reference data from the RoboPoint and RefSpatial datasets to enhance grounding accuracy.

All point coordinates are converted to absolute coordinates to align with the Qwen2.5-VL SmartResize prediction framework Bai et al. (2025b). Predicted coordinates are formatted in JSON and XML to support robust learning and adaptive processing of spatial instructions for diverse robotic tasks.

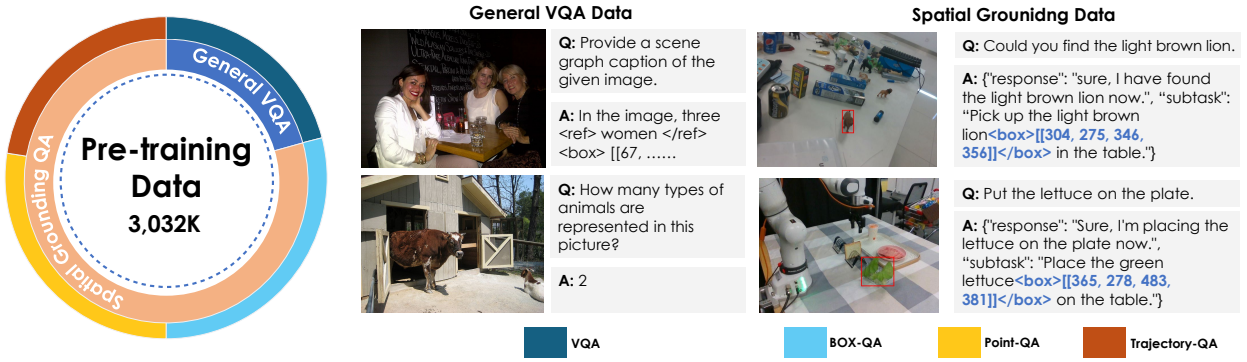


Figure 3. Overview of the pre-training data for the vision-language model. The data comprises two main parts: general VQA data to maintain the model's general multimodal capabilities, and spatial VQA data focusing on robotic-related grounding and spatial perception in a VQA format.

3.2. Synthetic Data For Action Post-Pre-training

To bridge the gap between VLM and VLA, we introduce a Post-Pre-Training phase, where large-scale simulated data is used to pre-train the VLA after VLM pre-training. This stage initializes the action head and facilitates the learning of action representations. Post-Pre-Training requires maintaining diversity both at the instruction and object levels. Consistent with the InternVLA-M1-Interface Data, we leverage GenManip as our data synthesis pipeline to construct a large-scale pick-and-place dataset, the InternData M1 dataset, which comprises 244K closed-loop samples. Specifically, we adopt the same object set and positional distributions as in InternVLA-M1-Interface Data, and process them through our scalable data pipeline. Each synthesized sample is rigorously validated to ensure correctness and consistency. To further enhance visual diversity, we introduce controlled randomization in lighting conditions and texture mappings.

3.3. Scalable Synthetic Data Engine for Instruction-Following

To support large-scale end-to-end data generation for VLM pre-training, we build a highly scalable, flexible, and fully automated simulation pipeline on top of GenManip Gao et al. (2025) and Isaac Sim Makoviychuk et al. (2021).

Automatic task synthesis for generalizable pick-and-place. We develop a scalable simulation pipeline (shown in Figure 4) that generates diverse manipulation trajectories from randomized object layouts and lighting conditions. By leveraging privileged simulation signals including object poses, object meshes, and robot arm state, the system rapidly generates scene layouts via a scene graph solver and computes candidate grasps based on object meshes Liang et al. (2019). Each candidate trajectory is then executed once in physics for closed-loop verification, after which a scene-graph validator checks whether the task goals are achieved. Only trajectories that both execute successfully and pass validation are accepted, ensuring that all collected data are physically feasible and task-complete.

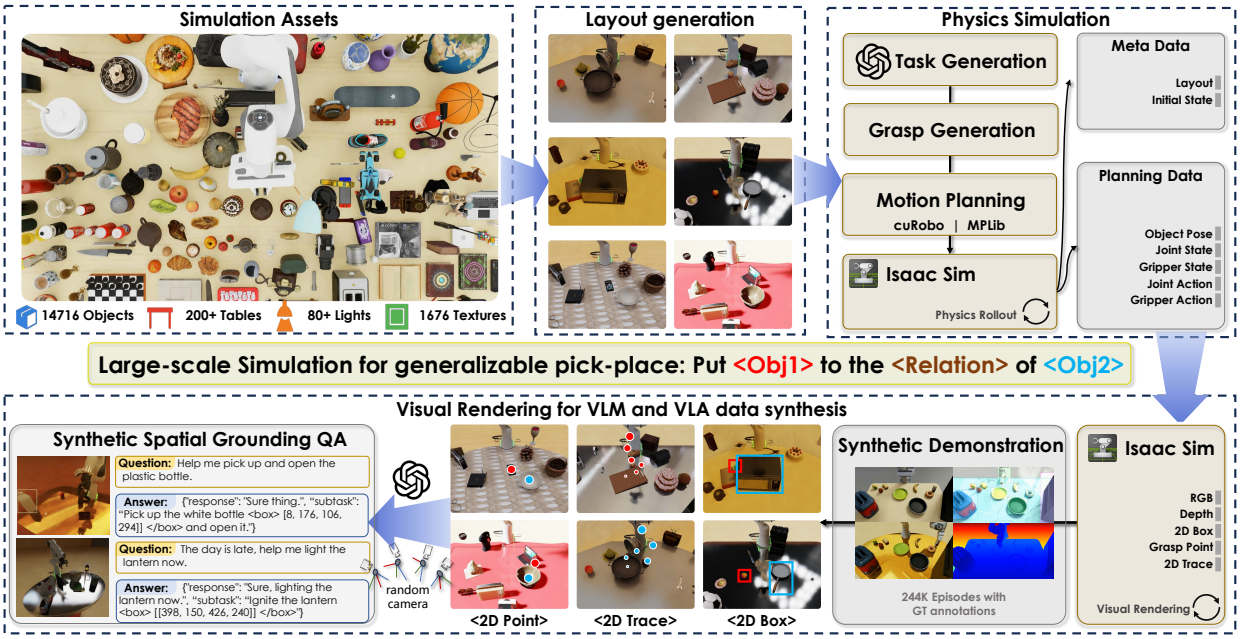


Figure 4. **Simulation data synthesis pipeline**. The pipeline generates diverse robotic manipulation data from a large asset library, converts intermediate representations into VQA data, and separates physics from rendering to reduce wasted failures and improve efficiency.

Synthesis of VLM data and VLA data for spatial grounding. For higher efficiency, robot planning and rendering are fully decoupled in our framework. The planner records structured scene and trajectory data, including joint states, object positions, and action information, which are later replayed by the renderer under randomized lighting, materials, and viewpoints. To align the simulation with real world, we calibrate all cameras using ArUco markers, ensuring that their intrinsic and extrinsic parameters match those of real-world cameras, thus maintaining consistent viewpoint geometry. In addition to high-resolution images, the renderer produces rich intermediate outputs, such as object bounding boxes and 2D end-effector trajectories. These signals provide dense supervision for action learning and facilitate the creation of auxiliary datasets for tasks such as spatial grounding, affordance reasoning, and trajectory prediction. Our asset library includes 14K annotated objects, 211 tables, 1.6K textures, and 87 dome lights, offering data with high visual and physical diversity—critical for developing generalizable models.

4. Experiments

We conducted extensive experiments to evaluate the performance of InternVLA-M1 in both simulation and real-world settings. First, we assess the performance on public simulated benchmarks (Section 4.1). Next, we fully evaluate the instruction-following of InternVLA-M1 for generalizable pick-and-place using Isaac-Sim (Section 4.2). Finally, we examine real-robot performance on long-horizon manipulation tasks to study instruction-following in real-world deployment (Section 4.2.2).

4.1. Experiments on Public Benchmarks

We use two established simulation suites:

- **SimplerEnv** is designed to probe robustness to visual appearance shifts. It includes both WidowX and Google Robot platforms, short-horizon atomic tasks, and controlled changes in lighting, color,

surface texture, and camera pose. We report results on three task sets: Google Robot-VM (visual matching under viewpoint and lighting changes), Google Robot-VA (visual aggregation with varying textures and colors), and WidowX-VM (cross-robot generalization).

- **LIBERO** is a language-conditioned manipulation suite built on a Franka arm with diverse scenes and expert demonstrations. We evaluate four task sets: LIBERO-Spatial (same objects, different spatial layouts), LIBERO-Object (fixed layout, different objects), LIBERO-Goal (fixed objects and layout, different goals), and LIBERO-Long (also known as LIBERO-10; longer tasks that span multiple objects, layouts, and operations).

Google Robot	Models	Co-Train	Pick Coke Can	Move Near	Open/Close Drawer	Open Top Drawer and Place Apple	Avg
Visual Matching	RT-1 Brohan et al. (2022)	✗	85.7	44.2	73.0	6.5	52.4
	RT-1-X Collaboration et al. (2023)	✗	56.7	31.7	59.7	21.3	42.4
	RT-2-X Brohan et al. (2023)	✓	78.7	77.9	25.0	3.7	46.3
	OpenVLA Kim et al. (2024)	✗	18.0	56.3	63.0	0.0	34.3
	CogACT Li et al. (2024c)	✗	91.3	85.0	71.8	50.9	74.8
	SpatialVLA Qu et al. (2025)	✗	86.0	77.9	57.4	-	75.1
	π_0 Black et al. (2024)	✗	72.7	65.3	38.3	-	58.8
	π_0-FAST Pertsch et al. (2025)	✗	75.3	67.5	42.9	-	61.9
	GR00T N1.5* Bjorck et al. (2025)	✗	51.7	54.0	27.8	7.4	35.2
	Magma Yang et al. (2025a)	✓	83.7	65.4	56.0	6.4	52.9
	Vanilla VLA	✗	90.0	69.8	52.5	52.2	66.1
	InternVLA-M1	✓	**95.3**	**90.0**	**75.5**	**62.0**	**80.7**
	Δ		+5.3	+20.2	+23.0	+9.8	+14.6
Variant Aggregation	RT-1 Brohan et al. (2022)	✗	89.8	50.0	32.3	2.6	43.7
	RT-1-X Collaboration et al. (2023)	✗	49.0	32.3	29.4	10.1	30.2
	RT-2-X Brohan et al. (2023)	✓	82.3	79.2	35.3	20.6	54.4
	OpenVLA Kim et al. (2024)	✗	60.8	67.7	28.8	0.0	39.3
	CogACT Li et al. (2024c)	✗	89.6	80.8	28.3	46.6	61.3
	SpatialVLA Qu et al. (2025)	✗	88.0	82.5	41.8	-	70.7
	π_0 Black et al. (2024)	✗	75.2	63.7	25.6	-	54.8
	π_0-FAST Pertsch et al. (2025)	✗	77.6	68.2	31.3	-	59.0
	GR00T N1.5 Bjorck et al. (2025)	✗	69.3	68.7	35.8	4.0	44.5
	Magma Yang et al. (2025a)	✓	68.8	65.7	53.4	18.5	51.6
	Vanilla VLA	✗	92.3	80.3	50.1	31.4	63.5
	InternVLA-M1	✓	**86.1**	**82.0**	**72.0**	**64.0**	**76.0**
	Δ		-6.2	+1.7	+21.9	+32.6	+12.5

Table 1. Result comparisons of robotic manipulation on SimplerEnv (Google-Robot) benchmark. The underlined scores indicate the best results excluding InternVLA-M1. Numbers are officially reported; otherwise, we reimplement and mark such entries with $*$.

Baselines. We compare to state-of-the-art open VLA systems, including π_0 Black et al. (2024), GR00T Bjorck et al. (2025), OpenVLA Kim et al. (2024), CogACT Li et al. (2024c), and etc. We also include a Vanilla VLA built on `QwenVL-2.5-3B-Instruct` with a DiT action head. When available, we use official reported numbers; otherwise, we reimplement and mark such entries with $*$. We keep training data, observation spaces, and action type aligned with the most popular setups Li et al. (2024c) to ensure a fair comparison.

4.1.1. SimplerEnv Benchmark

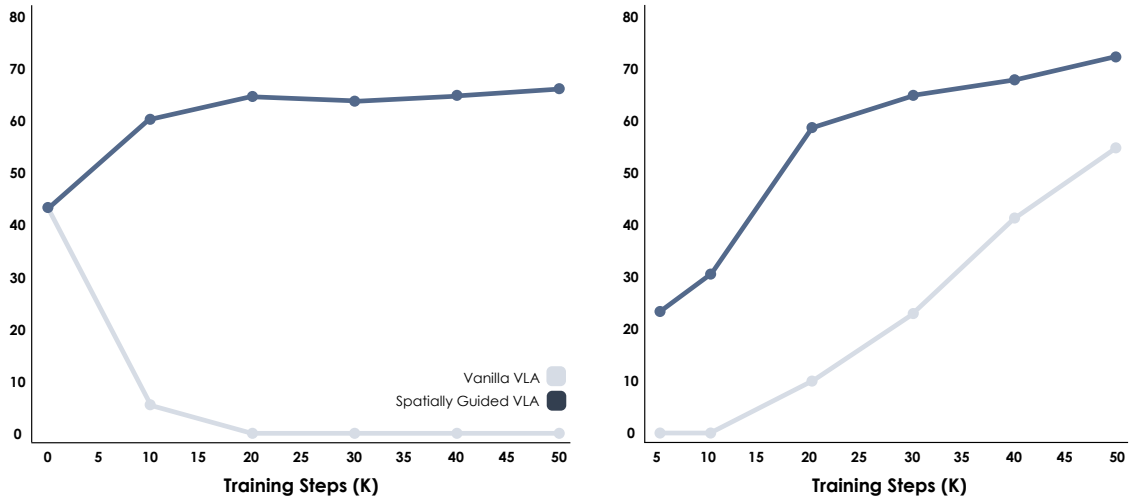
Experiment Setup. As described in Section 2.2, we post-train InternVLA-M1 on a subset of Open-X Embodiment (OXE) (including `fractal_rt_1` and `bridge_v1`), with co-training on spatial grounding data (Section 3.1). The VLM takes the primary observation image, task instruction, and an auxiliary spatial prompt as input, while the action expert predicts actions with an action chunk size

of 16. For multimodal data, the model follows an SFT-style question-answering format. Training is performed on 16 NVIDIA A100 GPUs for 50k steps (~2.5 epochs), with batch sizes of 16 for robot action data and 4 for multimodal data, optimized with a summed loss over both data types. All evaluations are conducted within SimplerEnv using its official evaluation protocol.

WidowX Robot	Models	Co-Train	Put Spoon on Towel	Put Carrot on Plate	Stack Green Block on Yellow Block	Put Eggplant in Yellow Basket	Avg
Visual Matching	RT-1-X Brohan et al. (2022)	✗	0.0	4.2	0.0	0.0	1.1
	Octo-Base Octo Model Team et al. (2024)	✗	15.8	12.5	0.0	41.7	17.5
	Octo-Small Octo Model Team et al. (2024)	✗	41.7	8.2	0.0	56.7	26.7
	OpenVLA Kim et al. (2024)	✗	4.2	0.0	0.0	12.5	4.2
	CogACT Li et al. (2024c)	✗	71.7	50.8	15.0	67.5	51.3
	SpatialVLA Qu et al. (2025)	✗	16.7	25.0	29.2	<u>100.0</u>	42.7
	π_0 Black et al. (2024)	✗	29.1	0.0	16.6	62.5	27.1
	π_0-FAST Pertsch et al. (2025)	✗	29.1	21.9	10.8	66.6	48.3
	GR00T N1.5 Bjorck et al. (2025)	✗	<u>75.3</u>	<u>54.3</u>	<u>57.0</u>	61.3	<u>61.9</u>
	Magma Yang et al. (2025a)	✓	37.5	31.0	12.7	60.5	35.8
	Vanilla VLA	✗	56.6	63.3	27.0	71.8	54.7
	InternVLA-M1	✓	**87.5**	**67.9**	**31.3**	**100.0**	**71.7**
	Δ		+30.9	+4.6	+4.3	+28.2	+17.0

Table 2. Result comparisons of robotic manipulation on SimplerEnv (WidowX) benchmark. The underlined scores indicate the best results excluding InternVLA-M1.

Result Analysis. The main experimental results are presented in Table 1 and Table 2. Compared with prior state-of-the-art models, it attains a 5.9% gain in Google Robot Visual Matching, a 5.3% gain in Visual Aggregation, and a 9.8% gain on the WidowX benchmark. These results highlight the strong competitiveness of InternVLA-M1 within the community. Compared to the Vanilla VLA based on QwenVL-2.5-3B-Instruct, InternVLA-M1 achieves substantial improvements: a 14.6% increase in Google Robot Visual Matching and a 12.4% increase in Visual Aggregation, along with a 17.0% improvement on the WidowX benchmark. These results demonstrate the effectiveness of our spatially guided pre-training and action post-training strategies.



(a) Manipulation performance (WindowX) across training steps

(b) Perception performance (RefCOCO+) acrosstraining steps.

Figure 5. Ablation study on the effect of Spatially Guided Action Post-training across different training steps. Figure 5 (a) evaluates spatial grounding performance on RefCOCO+ while Figure 5 (b) measures manipulation success rate on the WindowX task. *Input image resized to 224×224 to align with obs resolution in prior VLA work Black et al. (2024); Kim et al. (2024, 2025).

Ablation Study. Figure 5 presents a comparative analysis of manipulation performance (WindowX) and perception performance (RefCOCO+) across training steps. The results demonstrate that omitting spatial data and spatially guided prompting during training leads to rapid degradation of spatial grounding capabilities and slower convergence in manipulation tasks. In contrast, the proposed Spatially Guided Action Post-training method significantly accelerates convergence, substantially improves manipulation success rates, and enhances perceptual accuracy (the blue vs gray lines in Figure 5).

4.1.2. LIBERO Benchmark

Experimental Setups. Following Kim et al. (2025), we conduct experiments on the LIBERO benchmark. We filter out failed demonstrations and pause frames. During training, the policy takes as input both wrist-mounted and third-person camera views. We fine-tune the model on each suite independently using 8 A100 GPUs with a batch size of 128 and an action chunk size of 8. Training runs for roughly 30K steps, lasting about 20 hours. Each suite is evaluated with 500 trials.

Models	Spatial	Objects	Goal	Long	Avg
OpenVLA Kim et al. (2024)	84.7	88.4	79.2	53.7	76.5
SpatialVLA Qu et al. (2025)	88.2	89.9	78.6	55.5	78.1
CoT-VLA Zhao et al. (2025)	87.5	91.6	87.6	69.0	83.9
GR00T N1 Bjorck et al. (2025)	94.4	97.6	93.0	90.6	93.9
π_0 Black et al. (2024)	96.8	98.8	95.8	85.2	94.2
π_0-FAST Pertsch et al. (2025)	96.4	96.8	88.6	60.2	85.5
$\pi_{0.5}$-KI Driess et al. (2025)	98.0	97.8	95.6	85.8	94.3
Vanilla VLA	**98.8**	98.0	81.4	88.0	91.6
InternVLA-M1	98.0	**99.0**	**93.8**	**92.6**	**95.9**

Table 3. Result comparisons of robotic manipulation on LIBERO (Franka) benchmark.

Result analysis. The primary experimental results on the LIBERO benchmark are presented in Table 3. Compared to previous strong baselines, such as GR00T N1 and π_0, the InternVLA-M1 framework achieves notable improvements, particularly on the spatial and long-horizon tracks, with success rates of 98.0% and 92.6%, respectively. These results demonstrate the efficacy of our proposed method in managing complex, multi-step manipulation tasks. Specifically, for object placement, InternVLA-M1 attains a 99.0% SR, which highlights its robust object grounding capability.

4.2. Experiments on Instruction-Following in In-house Environment

4.2.1. Evaluation in Simulated Large-scale Pick-and-place

Existing benchmarks such as SimplerEnv and LIBERO are limited in scale, which restricts the comprehensive evaluation of instruction-following manipulation in diverse and cluttered settings. To more rigorously assess generalization capabilities, we conduct an experimental study on a large-scale simulation evaluation with enhanced object diversity and layout variation.

Experimental Setups. We constructed 200 pick-and-place tasks based on Isaac-Sim Gao et al. (2025), where the manipulated objects in each task are mutually distinct. Including background objects, the benchmark covers over 3K items and containers in total. Each task was executed once through the data generation pipeline to ensure its executability. Furthermore, for each of the 200 tasks, we additionally collected 5 trajectories with identical object sets but randomized layouts, which were

used for post-training. Both our model and all baseline models were trained using delta joint space control.

Result Analysis. As shown in Figure 6, we evaluate InternVLA-M1 under four generalization settings: In-Distribution, Unseen Objects, New Background, and Unseen Instructions. For each setting, we report two variants of the model: *w/o mid-train*, which is fine-tuned using only five trajectories per task, and *w/ mid-train*, which is additionally mid-trained on InternData M1 prior to fine-tuning. The results, summarized in Figure 7, show that across all settings, both variants outperform the baseline π_0, while InternVLA-M1 w/ mid-train consistently surpasses GR00T N1.5. Although InternVLA-M1 w/o mid-train exhibits slight variance in certain settings, the mid-trained variant achieves a consistent advantage, with an average gain of +6.2% over GR00T N1.5.

The performance on unseen objects highlights the benefit of simulation-enhanced visual generalization, enabling the model to handle novel instances beyond the training distribution. When evaluated under new backgrounds with randomized textures and layouts, both variants maintain strong performance, and the improvements from mid-training indicate increased robustness to scene-level shifts. Furthermore, under paraphrased instructions involving attribute-level or commonsense rewrites, InternVLA-M1 w/ mid-train demonstrates reliable instruction grounding, reflecting strong language generalization beyond templated expressions.

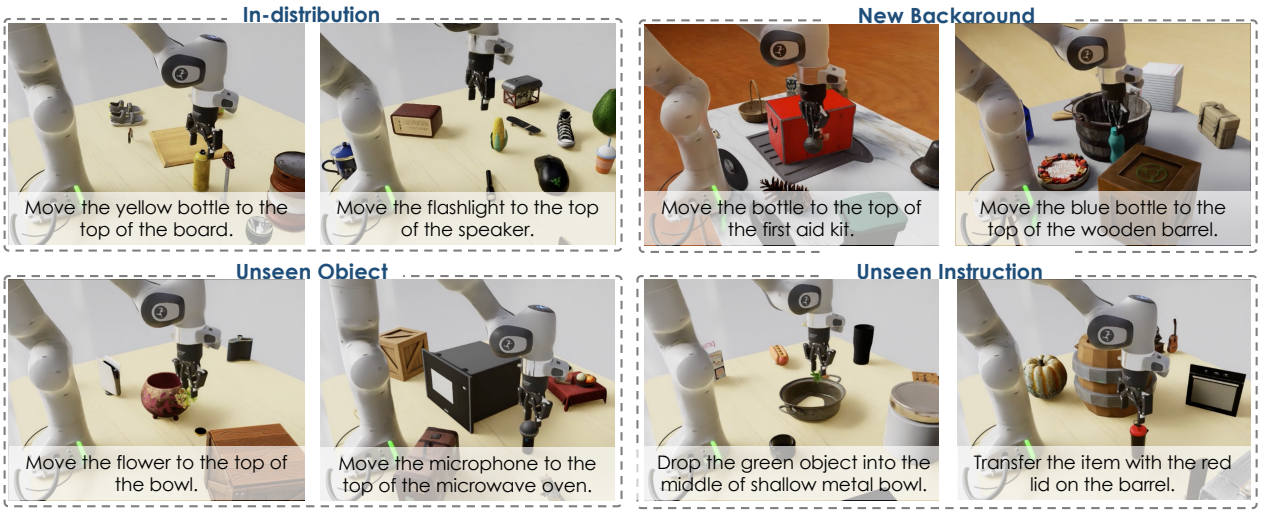


Figure 6. Evaluation settings for generalizable pick-and-place in large-scale simulation.

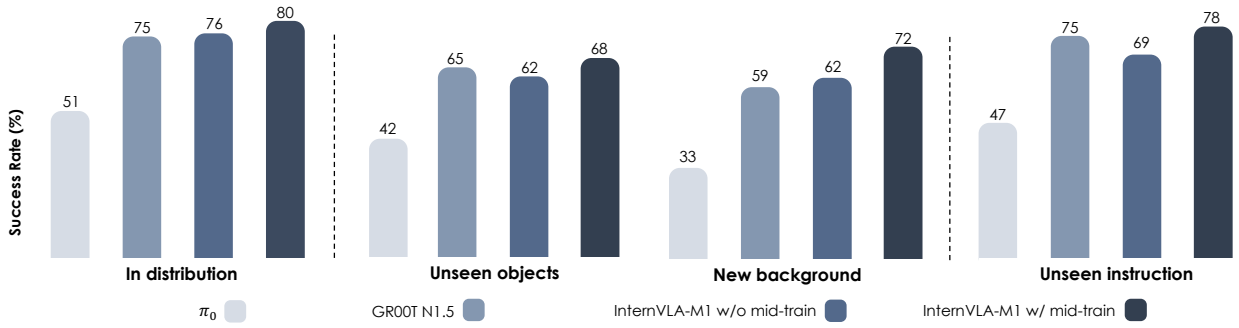


Figure 7. Result comparison of 200 simulated benchmarks in instruction-following pick-and-place.

4.2.2. Evaluation in Real-world Cluttered-scene Pick-and-Place

Experimental Setup. To evaluate the model's instruction-following capability on a large number of objects in real-world scenarios, we design a real-world object sorting benchmark consisting of single-horizon pick-and-place tasks. In this benchmark, a Franka Emika Panda robot performs pick-and-place operations within a 60×90 cm tabletop workspace. The benchmark includes 23 seen objects and 5 seen containers (detail listed in Figure 8). Three containers are randomly selected and placed at fixed tabletop locations, while a larger number of diverse objects are scattered randomly between them. The model is required to pick specified objects and place them into designated containers according to language instructions. To support model post-training, we collect 6 hours of demonstration data under this setting, where the dataset only contains objects and containers from a predefined "seen" set. We compare two variants of InternVLA-M1, *w/o co-train* and *w/ co-train*, against GR00T N1.5 and π_0 across five evaluation regimes on this benchmark. Here, *InternVLA-M1 w/o co-train* denotes fine-tuning *solely* on teleoperation-collected real-world demonstrations, with no simulation data. *InternVLA-M1 w/ co-train* additionally leverages the simulation dataset *InternData-M1* during training, together with the same real-world data. Both our model and all baseline models were trained using delta end-effector (EEF) space control in the real-world experiment.

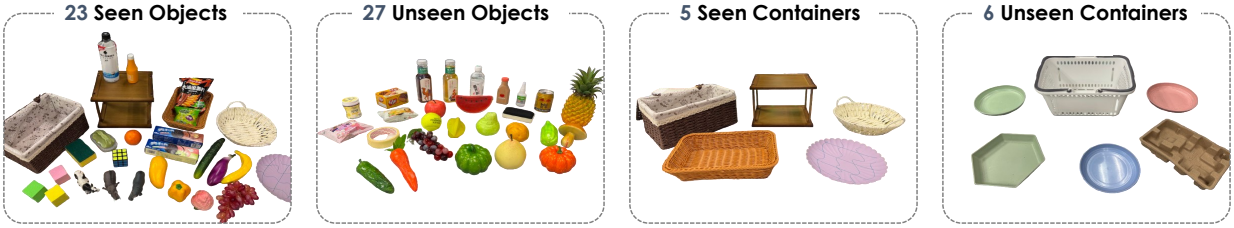


Figure 8. Overview of objects and containers used in instruction-following pick-and-place.

Evaluation Settings. To evaluate generalization, we further partition all available object and container assets into disjoint seen and unseen sets, as illustrated in Figure 8. Only the seen set is included in the training data, while both seen and unseen sets are evaluated during testing to measure the model's ability to generalize to novel objects. As shown in Figure 9, we evaluate instruction-following capabilities of various models on real-world pick-and-place tasks under the below settings: In-Distribution (ID), Unseen Objects (UO), Unseen Object Position (UOP), Unseen Object Orientation (UOO), and Unseen Instructions (UI). We report Success Rate (SR), defined as the fraction of trials in which the specified object is placed into the designated container. Higher SR indicates better performance. For each model, we conducted a total of 300 rollout evaluations. Each trial corresponds to one or more testing settings, and we ensured that each individual setting was evaluated at least 50 times. To ensure fair comparisons across models, we fixed the positions of the objects and containers for each task during testing.

Result Analysis. As shown in Figure 10, both variants of InternVLA-M1 demonstrate superior performance under the in-distribution setting, consistently outperforming GR00T N1.5 and π_0 when evaluated on objects and containers seen during training. This indicates strong instruction-following capabilities within familiar contexts. Beyond this, the inclusion of Interndata-M1 during co-training significantly enhances the model's visual generalization, enabling improved performance on novel objects not encountered during training. This suggests that synthetic data serves as an effective complement to limited real-world demonstrations. Additionally, because real-world data collection cannot exhaustively cover the spatial workspace, simulation data enriches the distribution of object positions and orientations. This leads to substantially better generalization to unseen configurations in terms of both object placement and pose. Finally, InternVLA-M1 maintains robust performance when

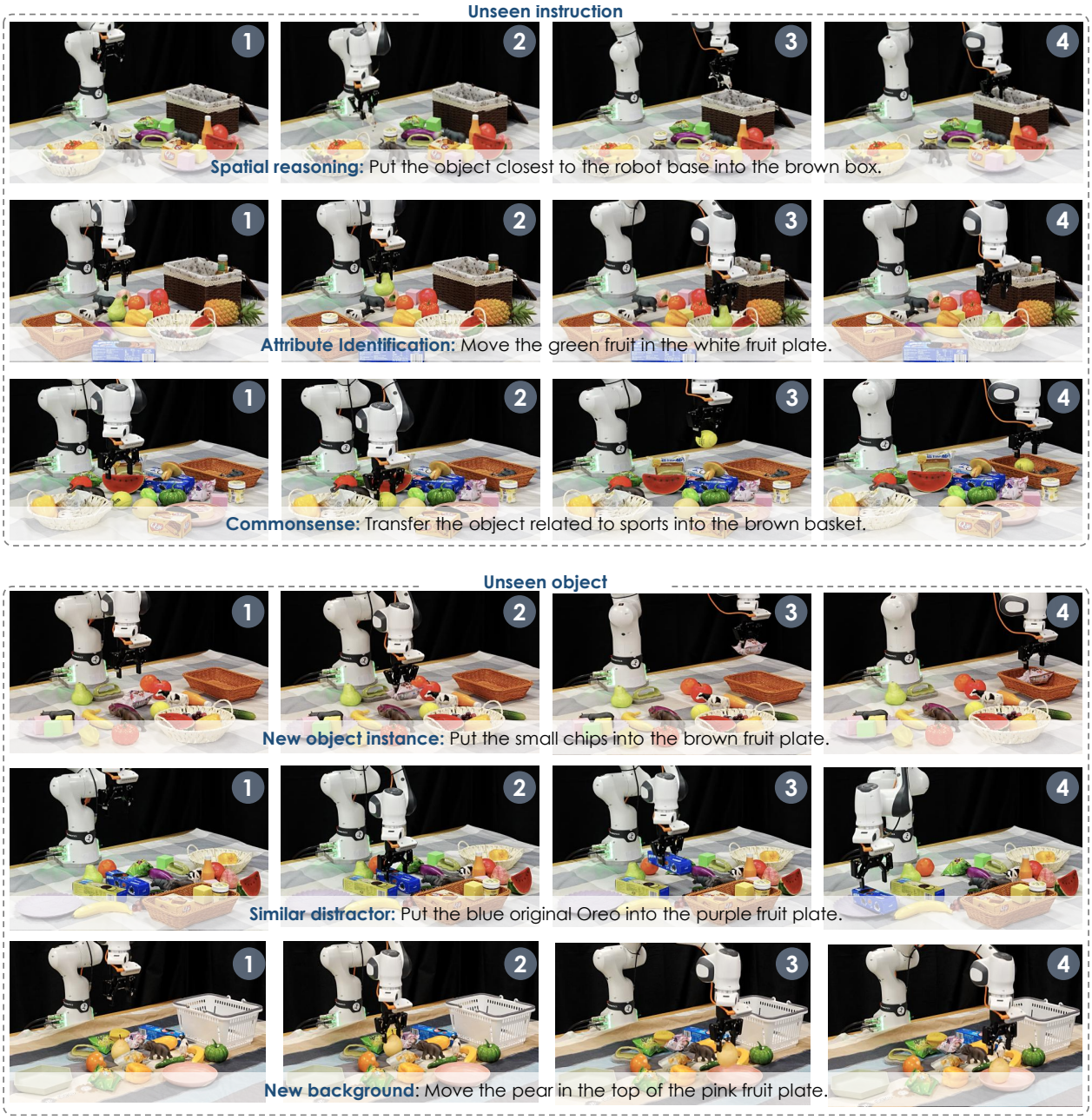


Figure 9. Evaluation settings showcase for real-world instruction-following manipulations.

given novel instructions, highlighting its ability to generalize across diverse linguistic expressions beyond those seen during training.

4.2.3. Evaluation in Long-horizon and Reasoning Manipulation

A key strength of our dual-system framework is its ability to leverage a high-level planner (System 2) to decompose long-horizon, reasoning-heavy tasks into a sequence of atomic actions, which are then robustly executed by a low-level action model (System 1). To evaluate this capability, we design a series of tasks that require not only multi-step planning but also the ability to reason about object attributes, monitor progress, and adapt to changes. As illustrated in Figure 11, these include:

- **Desktop Sorting.** The Franka robot is tasked with sorting objects into containers based on high-

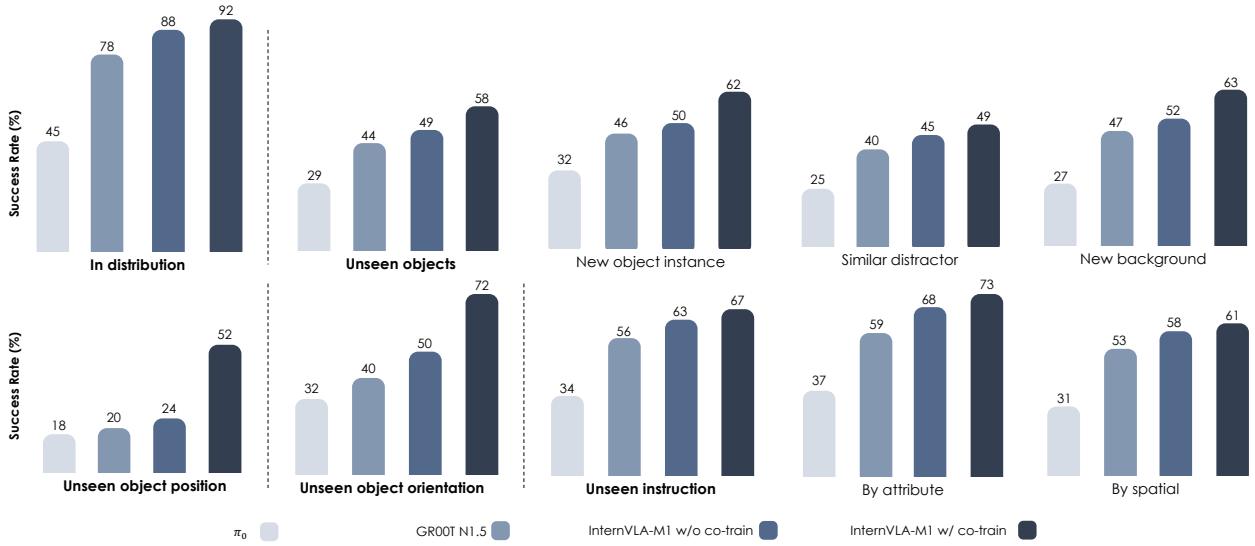


Figure 10. Result comparison in real-world instruction-following pick-and-place.

level semantic categories, aiming to ensure that all items on the desktop are eventually placed into the correct containers. Both objects and containers are scattered within a 60×90 cm region in front of the robot base. The setup includes five seen containers and five object categories: *fruits, toys, vegetables, bottles*, and *snacks*. Each evaluation instance involves sorting objects from one to three categories into their respective containers. Each trial consists of three pick-and-place actions, and we report success rates consistent with the metric used for pick-and-place under clustered environments.

- **Sorting Items into Drawers.** The Franka robot is required to (i) open a designated drawer (either lower or upper), (ii) place the target objects into it, and (iii) close the drawer. This task demands precise temporal reasoning and articulated manipulation. The objects are placed within a 35×35 cm area located to the front-right of the robot base. We report stepwise execution success, where a step is considered valid only if all preceding steps have succeeded.
- **Making Sandwiches.** The Franka robot is instructed to assemble sandwiches following a predefined meal recipe. Ingredients and plates are placed within a 50×70 cm region in front of the robot base. We define five types of sandwich recipes as the seen set: [bread–lettuce–bread], [bread–lettuce–meat–bread], [bread–meat–lettuce–meat–bread], [bread–meat–meat–bread], and [bread–meat–bread]. We report success rates on both the seen set and an unseen set involving real-time environment interaction, using the same success definition as in the drawer sorting task.
- **Math Calculation.** The Franka robot is prompted to solve a math problem and press the color-coded button (red, yellow, or blue) that corresponds to the correct answer based on arithmetic reasoning. The buttons are randomly placed within a 40×40 cm area in front of the robot base.
- **Goods Purchase.** The ARX LIFT2 dual-arm robot is tasked with identifying and placing into a basket the object bearing the correct price tag, given a numerical cue ranging from 1 to 9. We report the success rate of correctly placing the item corresponding to the queried price into the basket.

Experimental Setup. To support fine-grained training for these long-horizon tasks, we collect a total of **22** hours of high-quality long-horizon and reasoning teleoperated demonstrations, amounting to approximately 400–500 trajectories per task. Each collected trajectory is segmented into *subtasks* and annotated with corresponding atomic actions. For example, a “make a classic sandwich” task is decomposed into four subtasks: (1) “Put a piece of bun on the plate.” → (2) “Put a piece of meat

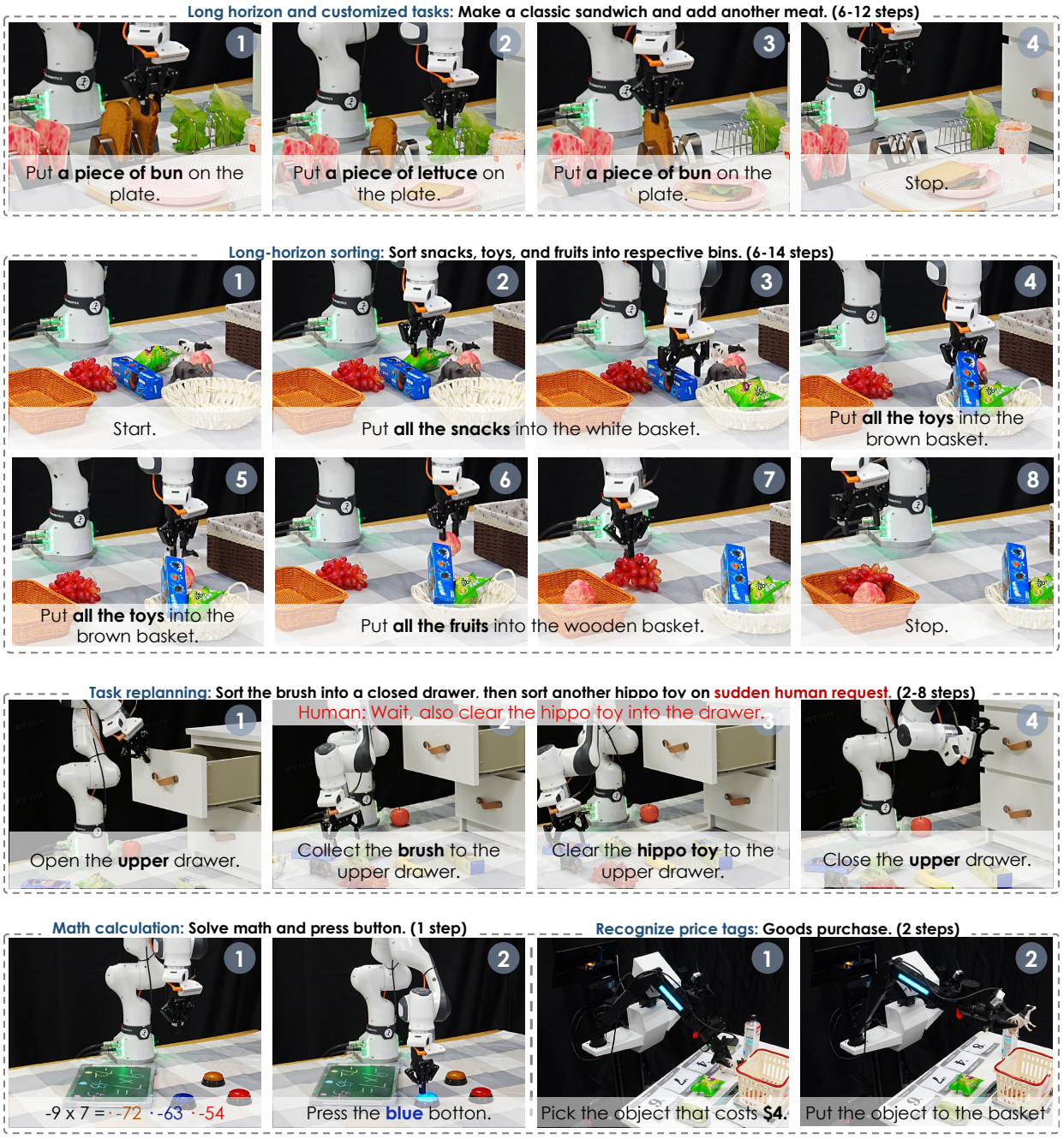


Figure 11. Showcase for long-horizon instruction-following manipulation.

on the plate." → (3) "Put a piece of lettuce on the plate." → (4) "Put a piece of bun on the plate." Each sub-instruction is paired with a specific segment of the demonstration. To enable subtask-level transition, we introduce zero-action vectors padding after each subtask segment. This allows the model to stop upon subtask completion and then be prompted to predict the transition to the next subtask. In addition, to improve temporal consistency and ensure smooth inference, we remove frames in which the robot arm exhibits clear pauses or idle behavior. In contrast to prior VLA models that depend on an additional VLM to serve as a task planner for long-horizon or reasoning-intensive tasks, our unified model architecture is trained jointly on multimodal inputs encompassing task decomposition, subtask identification, numerical reasoning, and action supervision. This joint training paradigm

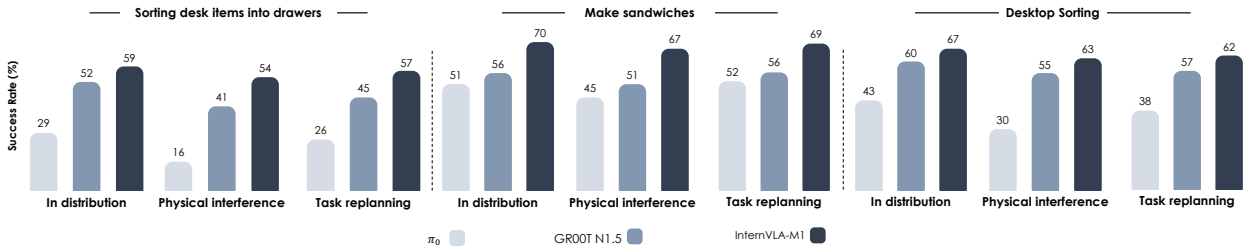


Figure 12. Result comparison in real-world long-horizon task planning for manipulation.

enables a single model to seamlessly integrate task planning, reasoning, and action prediction in an end-to-end fashion. As shown in Table Table 4, compared to prior VLM-based approaches, our unified model provides more effective task decomposition and numerical reasoning for both long-horizon and reasoning-intensive tasks.

Models	Long-horizon tasks			Reasoning tasks	
	Sort into Drawers	Make Sandwiches	Desktop Sorting	Math calculation	Goods Purchase
Gemini-2.5 Pro	57	62	<u>83</u>	53	61
GPT-5	<u>75</u>	67	62	<u>79</u>	<u>82</u>
GPT-4o	37	57	35	39	41
Qwen2.5-VL-72B	31	<u>71</u>	34	33	29
Qwen2.5-VL-3B	30	49	52	41	38
Ours-3B	**90**	**91**	**91**	**93**	**92**

Table 4. Task scheduling performance of VLM planner in long-horizon and reasoning scenarios.

Evaluation Settings. We evaluate model performance under three distinct settings, In-distribution, Physical Interference and Task Replanning:

- **Physical Interference.** External disturbances are introduced during task execution. For example, during the *sorting items into drawers* task, the drawer is manually closed after the robot opens it, or the target object is displaced during grasping. This evaluates the model's ability to perceive environmental changes and adapt accordingly.
- **Task Replanning.** New instructions are issued mid-execution. For instance, after placing an object in the drawer but before closing it, the robot is told: "Also put the cow toy into the top drawer." This tests the model's ability to incorporate new subgoals and dynamically adjust its plan.

Results Analysis. As shown in Figure 12, across long horizon tasks, InternVLA-M1 consistently outperforms the baselines, enabled by its unified subtask planning mechanism. In the in distribution setting, it achieves more reliable execution than GR00T N1.5 and π_0, showing stronger grounding of high level goals into actionable steps. Under physical interference, the model demonstrates robust adaptability: for example, in desktop sorting when containers are unexpectedly moved, InternVLA-M1 can track the new container locations and complete the placement. Moreover, when task replanning is required, such as when additional instructions are introduced during execution, InternVLA-M1 is able to revise its subtask sequence on the fly and continue with correct actions. This adaptability leads to minimal performance degradation under stress conditions, while the baselines exhibit much larger declines, underscoring the model's resilience to dynamic environments and shifting instructions.

5. Related work

Hierarchical Robot System. A key challenge in embodied AI is bridging high-level instructions with low-level actions, a problem often addressed by generating intermediate representations (IRs) that exist on a spectrum from formal symbolic structures to learned embeddings Xie et al. (2019). Inspired by Chain-of-Thought (CoT) reasoning, many approaches train vision-language-action (VLA) models to generate explicit textual plans before acting, which enhances interpretability and performance on complex tasks Zawalski et al. (2024). Beyond textual plans, research has explored more structured or physically grounded IRs. Historically, many systems relied on more direct perceptual outputs as IRs, such as bounding boxes from object detectors for manipulation Griffin (2023), specific 3D points for grasp planning from point clouds Ten Pas and Platt (2017), or dense correspondence fields derived from self-supervised features learned for control, like DINO features Laskin et al. (2020); Nair et al. (2022). Some systems build persistent 3D scene graphs as a comprehensive world model that an LLM can query to ground long-horizon plans Rana et al. (2023). Others focus on action-centric IRs, such as conditioning policies on visual affordances that specify the robot's end-effector pose at key moments in a task Nasiriany et al. (2024). A noticeable trend is the generation of explicit spatial localizers that are directly consumable by robot controllers Gu et al. (2023); Huang et al. (2025b); Li et al. (2025c). Large-scale foundation models Luo et al. (2025); Team et al. (2025) are trained to unify perception and planning, outputting not just plans but also affordance predictions as bounding boxes. For tasks requiring higher precision, specialized models like RoboRefer Zhou et al. (2025a) use dedicated architectures and reinforcement learning to predict exact 3D coordinates from complex spatial language. In contrast to specialized system 2 models that generate intermediate representations, our model is more unified. It can not only output these intermediate representations but also directly utilize them in downstream VLA tasks. This unification allows for an integrated, end-to-end training process, enabling the direct adaptation of training strategies through feedback from real-world deployment.

Vision-Language-Action Models with Explicit Language Reasoning. Chain-of-Thought prompting has proven effective for improving reasoning in Large Language Models Wei et al. (2022), and its success has inspired extensions to embodied AI. In Vision-Language-Action (VLA) models, generating intermediate reasoning steps before acting enables agents to handle complex, long-horizon tasks. These methods can be grouped by reasoning modality. Early approaches emphasized linguistic reasoning. ECOT Zawalski et al. (2024) elicits explicit text-based plans and sub-tasks to enhance performance and interpretability. RT-H Belkhale et al. (2024) introduces a fine-grained "action language" for hierarchical policies and human intervention, while InstructVLA Yang et al. (2025b) jointly optimizes reasoning and action through VLA-IT, improving generalization. OneTwoVLA Lin et al. (2025) adaptively alternates between "thinking" and execution, RAD Clark et al. (2025) leverages action-free human videos to derive reasoning guides, and $\pi_{0.5}$ Intelligence et al. (2025) trains on heterogeneous data before fine-tuning for subtask prediction. To better connect reasoning with the physical world, later work shifted toward visual and spatial modalities. GraphCoT-VLA Huang et al. (2025a), for example, employs a dynamic pose-object graph for 3D spatial reasoning. Despite their differences, these approaches share a common feature: they explicitly generate intermediate steps—textual, visual, or spatial—during inference. While effective, this incurs additional computational cost. In contrast, we propose a post-training phase that directly unlocks the VLM's intrinsic reasoning capacity, eliminating the need for explicit generative reasoning. Our model matches the performance of CoT-based methods while avoiding token overhead, showing that strong reasoning can be elicited without costly intermediate outputs.

Generalist Robot Policy. Recent research in general-purpose robotics has seen the emergence of several mainstream technical paradigms. Monolithic VLA models utilize a single end-to-end network

to directly map multimodal inputs to tokenized low-level actions, as demonstrated by systems Brohan et al. (2023); Kim et al. (2024); Lee et al. (2025); Yang et al. (2025a). In contrast, unified architectures decouple high-level cognition from low-level action, allowing for greater modularity and interpretability. This category has seen extensive exploration Black et al. (2024); Li et al. (2025a, 2024c) leveraging specialized generative models for action synthesis. Other notable approaches in this vein Cheang et al. (2025); Intelligence et al. (2025); Shukor et al. (2025); Song et al. (2025); Yang et al. (2025b); Zhou et al. (2025b), which uses an LLM to break down high-level language commands into intermediate action plans. A third paradigm is based on world models, which learn a predictive model of the environment's dynamics to enable planning and control. These models allow for simulating future outcomes, often facilitating planning via search in a learned latent space or by conditioning a separate policy. While powerful, this approach can be computationally intensive. Representative works Bjorck et al. (2025); Bu et al. (2025b); Cen et al. (2025); Li et al. (2025b); Liao et al. (2025); Lv et al. (2025); Tian et al. (2024); Wang et al. (2025); Ye et al. (2025) exemplify this forward-predictive approach to decision-making. Our model adopts a typical dual-system approach, building upon the VLA with unified architectures, then introducing additional planning design, thereby achieving better adaptability to real-world environments.

6. Discussion and conclusion

In this work, we presented InternVLA-M1, a unified vision-language-action framework that leverages spatial grounding priors to bridge high-level multimodal reasoning with low-level robotic execution. By combining large-scale multimodal pre-training with spatially guided post-training, our model effectively transfers perceptual and reasoning skills into embodied control, achieving strong generalization to unseen objects, instructions, and environments. Extensive evaluations across simulation and real-world settings demonstrate that InternVLA-M1 surpasses existing VLA models and specialized systems in instruction following, long-horizon manipulation, and multimodal grounding, highlighting spatial reasoning as a unifying substrate for scalable and reliable generalist robots.

References

F. AI. Helix, 2024. URL https://www.figure.ai/news/helix.

S. Bai, K. Chen, X. Liu, J. Wang, W. Ge, S. Song, K. Dang, P. Wang, S. Wang, J. Tang, H. Zhong, Y. Zhu, M. Yang, Z. Li, J. Wan, P. Wang, W. Ding, Z. Fu, Y. Xu, J. Ye, X. Zhang, T. Xie, Z. Cheng, H. Zhang, Z. Yang, H. Xu, J. Lin, and ... Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025a.

S. Bai, K. Chen, X. Liu, J. Wang, W. Ge, S. Song, K. Dang, P. Wang, S. Wang, J. Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025b.

S. Belkhale, T. Ding, T. Xiao, P. Sermanet, Q. Vuong, J. Tompson, Y. Chebotar, D. Dwibedi, and D. Sadigh. Rt-h: Action hierarchies using language. *arXiv preprint arXiv:2403.01823*, 2024.

J. Bjorck, F. Castañeda, N. Cherniadev, X. Da, R. Ding, L. Fan, Y. Fang, D. Fox, F. Hu, S. Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, et al. $\backslash pi_0$: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

Q. Bu, J. Cai, L. Chen, X. Cui, Y. Ding, S. Feng, S. Gao, X. He, X. Hu, X. Huang, et al. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. *arXiv preprint arXiv:2503.06669*, 2025a.

Q. Bu, Y. Yang, J. Cai, S. Gao, G. Ren, M. Yao, P. Luo, and H. Li. Univla: Learning to act anywhere with task-centric latent actions. *arXiv preprint arXiv:2505.06111*, 2025b.

M. Cao, B. R. Team, et al. Robobrain 2.0 technical report. Technical report, Beijing Academy of Artificial Intelligence (BAAI), 2025. arXiv preprint arXiv:2507.02029.

J. Cen, C. Yu, H. Yuan, Y. Jiang, S. Huang, J. Guo, X. Li, Y. Song, H. Luo, F. Wang, et al. Worldvla: Towards autoregressive action world model. *arXiv preprint arXiv:2506.21539*, 2025.

C. Cheang, S. Chen, Z. Cui, Y. Hu, L. Huang, T. Kong, H. Li, Y. Li, Y. Liu, X. Ma, et al. Gr-3 technical report. *arXiv preprint arXiv:2507.15493*, 2025.

Z. Chen, J. Wu, W. Wang, W. Su, G. Chen, S. Xing, M. Zhong, Q. Zhang, X. Zhu, L. Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198, 2024.

C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Proceedings of Robotics: Science and Systems (RSS)*, 2023.

J. Clark, S. Mirchandani, D. Sadigh, and S. Belkhale. Action-free reasoning for policy generalization. *arXiv preprint arXiv:2502.03729*, 2025.

O. X.-E. Collaboration, A. O'Neill, A. Rehman, A. Gupta, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain, A. Tung, A. Bewley, A. Herzog, A. Irpan, A. Khazatsky, A. Rai, A. Gupta, A. Wang, A. Kolobov, A. Singh, A. Garg, A. Kembhavi, A. Xie, A. Brohan, A. Raffin, A. Sharma, A. Yavary, A. Jain, A. Balakrishna, A. Wahid, B. Burgess-Limerick, B. Kim, B. Schölkopf, B. Wulfe, B. Ichter, C. Lu, C. Xu, C. Le, C. Finn, C. Wang, C. Xu, C. Chi, C. Huang, C. Chan, C. Agia, C. Pan, C. Fu, C. Devin, D. Xu, D. Morton, D. Driess, D. Chen, D. Pathak, D. Shah, D. Büchler, D. Jayaraman, D. Kalashnikov, D. Sadigh, E. Johns, E. Foster, F. Liu, F. Ceola, F. Xia, F. Zhao, F. V. Frujeri, F. Stulp, G. Zhou, G. S. Sukhatme, G. Salhotra, G. Yan, G. Feng, G. Schiavi, G. Berseth, G. Kahn, G. Yang, G. Wang, H. Su, H.-S. Fang, H. Shi, H. Bao, H. B. Amor, H. I. Christensen, H. Furuta, H. Bharadhwaj, H. Walke, H. Fang, H. Ha, I. Mordatch, I. Radosavovic, I. Leal, J. Liang, J. Abou-Chakra, J. Kim, J. Drake, J. Peters, J. Schneider, J. Hsu, J. Vakil, J. Bohg, J. Bingham, J. Wu, J. Gao, J. Hu, J. Wu, J. Wu, J. Sun, J. Luo, J. Gu, J. Tan, J. Oh, J. Wu, J. Lu, J. Yang, J. Malik, J. Silvério, J. Hejna, J. Booher, J. Tompson, J. Yang, J. Salvador, J. J. Lim, J. Han, K. Wang, K. Rao, K. Pertsch, K. Hausman, K. Go, K. Gopalakrishnan, K. Goldberg, K. Byrne, K. Oslund, K. Kawaharazuka, K. Black, K. Lin, K. Zhang, K. Ehsani, K. Lekkala, K. Ellis, K. Rana, K. Srinivasan, K. Fang, K. P. Singh, K.-H. Zeng, K. Hatch, K. Hsu, L. Itti, L. Y. Chen, L. Pinto, L. Fei-Fei, L. Tan, L. J. Fan, L. Ott, L. Lee, L. Weihs, M. Chen, M. Lepert, M. Memmel, M. Tomizuka, M. Itkina, M. G. Castro, M. Spero, M. Du, M. Ahn, M. C. Yip, M. Zhang, M. Ding, M. Heo, M. K. Srirama, M. Sharma, M. J. Kim, N. Kanazawa, N. Hansen, N. Heess, N. J. Joshi, N. Suenderhauf, N. Liu, N. D. Palo, N. M. M. Shafiullah, O. Mees, O. Kroemer, O. Bastani, P. R. Sanketi, P. T. Miller, P. Yin, P. Wohlhart, P. Xu, P. D. Fagan, P. Mitrano, P. Sermanet, P. Abbeel, P. Sundaresan, Q. Chen, Q. Vuong, R. Rafailov, R. Tian, R. Doshi, R. Mart'in-Mart'in, R. Baijal, R. Scalise, R. Hendrix, R. Lin, R. Qian, R. Zhang, R. Mendonca, R. Shah, R. Hoque, R. Julian, S. Bustamante, S. Kirmani, S. Levine, S. Lin, S. Moore, S. Bahl, S. Dass, S. Sonawani, S. Tulsiani, S. Song, S. Xu, S. Haldar, S. Karamcheti, S. Adebola, S. Guist, S. Nasiriany, S. Schaal, S. Welker, S. Tian, S. Ramamoorthy, S. Dasari, S. Belkhale, S. Park, S. Nair, S. Mirchandani, T. Osa, T. Gupta, T. Harada, T. Matsushima, T. Xiao, T. Kollar, T. Yu, T. Ding, T. Davchev, T. Z. Zhao, T. Armstrong, T. Darrell, T. Chung, V. Jain, V. Kumar, V. Vanhoucke, W. Zhan, W. Zhou, W. Burgard, X. Chen, X. Chen, X. Wang, X. Zhu, X. Geng, X. Liu, X. Liangwei, X. Li, Y. Pang, Y. Lu, Y. J. Ma, Y. Kim, Y. Chebotar, Y. Zhou, Y. Zhu, Y. Wu, Y. Xu, Y. Wang, Y. Bisk, Y. Dou, Y. Cho, Y. Lee, Y. Cui, Y. Cao, Y.-H. Wu, Y. Tang, Y. Zhu, Y. Zhang, Y. Jiang, Y. Li, Y. Li, Y. Iwasawa, Y. Matsuo, Z. Ma, Z. Xu, Z. J. Cui, Z. Zhang, Z. Fu, and Z. Lin. Open X-Embodiment: Robotic learning datasets and RT-X models. https://arxiv.org/abs/2310.08864, 2023.

M. Deitke, C. Clark, S. Lee, R. Tripathi, Y. Yang, J. S. Park, M. Salehi, N. Muennighoff, K. Lo, L. Soldaini, J. Lu, T. Anderson, E. Bransom, K. Ehsani, H. Ngo, Y. Chen, A. Patel, M. Yatskar, C. Callison-Burch, A. Head, R. Hendrix, F. Bastani, E. VanderBilt, N. Lambert, Y. Chou, A. Chheda, J. Sparks, S. Skjonsberg, M. Schmitz, A. Sarnat, B. Bischoff, P. Walsh, C. Newell, P. Wolters, T. Gupta, K.-H. Zeng, J. Borchardt, D. Groeneveld, J. Dumas, C. Nam, S. Lebrecht, C. Wittlif, C. Schoenick, O. Michel, R. Krishna, L. Weihs, N. A. Smith, H. Hajishirzi, R. Girshick, A. Farhadi, and A. Kembhavi. Molmo and pixmo: Open weights and open data for state-of-the-art multimodal models. *arXiv preprint arXiv:2409.17146*, 2024.

D. Driess, J. T. Springenberg, B. Ichter, L. Yu, A. Li-Bell, K. Pertsch, A. Z. Ren, H. Walke, Q. Vuong, L. X. Shi, et al. Knowledge insulating vision-language-action models: Train fast, run fast, generalize better. *arXiv preprint arXiv:2505.23705*, 2025.

H.-S. Fang, C. Wang, H. Fang, M. Gou, J. Liu, H. Yan, W. Liu, Y. Xie, and C. Lu. Anygrasp: Robust and efficient grasp perception in spatial and temporal domains. *IEEE Transactions on Robotics*, 2023.

N. Gao, Y. Chen, S. Yang, X. Chen, Y. Tian, H. Li, H. Huang, H. Wang, T. Wang, and J. Pang. Genmanip: Llm-driven simulation for generalizable instruction-following manipulation. In *CVPR*, 2025.

B. Griffin. Mobile robot manipulation using pure object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 561–571, 2023.

J. Gu, S. Kirmani, P. Wohlhart, Y. Lu, M. G. Arenas, K. Rao, W. Yu, C. Fu, K. Gopalakrishnan, Z. Xu, et al. Rt-trajectory: Robotic task generalization via hindsight trajectory sketches. *arXiv preprint arXiv:2311.01977*, 2023.

H. Huang, F. Lin, Y. Hu, S. Wang, and Y. Gao. Copa: General robotic manipulation through spatial constraints of parts with foundation models. *arXiv preprint arXiv:2403.08248*, 2024a.

H. Huang, M. Cen, K. Tan, X. Quan, G. Huang, and H. Zhang. Graphcot-vla: A 3d spatial-aware reasoning vision-language-action model for robotic manipulation with ambiguous instructions. *arXiv preprint arXiv:2508.07650*, 2025a.

H. Huang, X. Chen, Y. Chen, H. Li, X. Han, Z. Wang, T. Wang, J. Pang, and Z. Zhao. Roboground: Robotic manipulation with grounded vision-language priors. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22540–22550, 2025b.

W. Huang, C. Wang, R. Zhang, Y. Li, J. Wu, and L. Fei-Fei. Voxposer: Composable 3d value maps for robotic manipulation with language models. *arXiv preprint arXiv:2307.05973*, 2023.

W. Huang, C. Wang, Y. Li, R. Zhang, and L. Fei-Fei. Rekep: Spatio-temporal reasoning of relational keypoint constraints for robotic manipulation. *arXiv preprint arXiv:2409.01652*, 2024b.

P. Intelligence, K. Black, N. Brown, J. Darpinian, K. Dhabalia, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, et al. $pi_{0.5}$: a vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025.

A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, S. Dasari, S. Karamcheti, S. Nasiriany, M. K. Srirama, L. Y. Chen, K. Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

M. J. Kim, C. Finn, and P. Liang. Fine-tuning vision-language-action models: Optimizing speed and success. *arXiv preprint arXiv:2502.19645*, 2025.

A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick. Segment anything. *arXiv:2304.02643*, 2023.

M. Laskin, A. Srinivas, and P. Abbeel. Curl: Contrastive unsupervised representations for reinforcement learning. In *International conference on machine learning*, pages 5639–5650. PMLR, 2020.

J. Lee, J. Duan, H. Fang, Y. Deng, S. Liu, B. Li, B. Fang, J. Zhang, Y. R. Wang, S. Lee, et al. Molmoact: Action reasoning models that can reason in space. *arXiv preprint arXiv:2508.07917*, 2025.

B. Li, Y. Zhang, D. Guo, R. Zhang, F. Li, H. Zhang, K. Zhang, P. Zhang, Y. Li, Z. Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024a.

B. Li, Y. Zhang, D. Guo, R. Zhang, F. Li, H. Zhang, K. Zhang, P. Zhang, Y. Li, Z. Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024b.

H. Li, S. Yang, Y. Chen, Y. Tian, X. Yang, X. Chen, H. Wang, T. Wang, F. Zhao, D. Lin, et al. Cronusvla: Transferring latent motion across time for multi-frame prediction in manipulation. *arXiv preprint arXiv:2506.19816*, 2025a.

Q. Li, Y. Liang, Z. Wang, L. Luo, X. Chen, M. Liao, F. Wei, Y. Deng, S. Xu, Y. Zhang, et al. Cogact: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation. *arXiv preprint arXiv:2411.19650*, 2024c.

S. Li, Y. Gao, D. Sadigh, and S. Song. Unified video action model. *arXiv preprint arXiv:2503.00200*, 2025b.

Y. Li, Y. Deng, J. Zhang, J. Jang, M. Memmel, R. Yu, C. R. Garrett, F. Ramos, D. Fox, A. Li, et al. Hamster: Hierarchical action models for open-world robot manipulation. *arXiv preprint arXiv:2502.05485*, 2025c.

H. Liang, X. Ma, S. Li, M. Görner, S. Tang, B. Fang, F. Sun, and J. Zhang. Pointnetgpd: Detecting grasp configurations from point sets. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 3629–3635. IEEE, 2019.

Y. Liao, P. Zhou, S. Huang, D. Yang, S. Chen, Y. Jiang, Y. Hu, J. Cai, S. Liu, J. Luo, et al. Genie envisioner: A unified world foundation platform for robotic manipulation. *arXiv preprint arXiv:2508.05635*, 2025.

F. Lin, R. Nai, Y. Hu, J. You, J. Zhao, and Y. Gao. Onetwovla: A unified vision-language-action model with adaptive reasoning. *arXiv preprint arXiv:2505.11917*, 2025.

F. Liu, K. Fang, P. Abbeel, and S. Levine. Moka: Open-vocabulary robotic manipulation through mark-based visual prompting. In *First Workshop on Vision-Language Models for Navigation and Manipulation at ICRA 2024*, 2024.

Y. Lu, Y. Fan, B. Deng, F. Liu, Y. Li, and S. Wang. Vl-grasp: a 6-dof interactive grasp policy for language-oriented objects in cluttered indoor scenes. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 976–983. IEEE, 2023.

G. Luo, G. Yang, Z. Gong, G. Chen, H. Duan, E. Cui, R. Tong, Z. Hou, T. Zhang, Z. Chen, et al. Visual embodied brain: Let multimodal large language models see, think, and control in spaces. *arXiv preprint arXiv:2506.00123*, 2025.

Q. Lv, W. Kong, H. Li, J. Zeng, Z. Qiu, D. Qu, H. Song, Q. Chen, X. Deng, M. Y. Wang, L. Nie, and J. Pang. F1: A vision-language-action model bridging understanding and generation to actions. 2025. URL https://arxiv.org/abs/2509.06951.

V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, et al. Isaac gym: High performance gpu-based physics simulation for robot learning. *arXiv preprint arXiv:2108.10470*, 2021.

J. Mao, J. Huang, A. Toshev, O. Camburu, A. L. Yuille, and K. Murphy. Generation and comprehension of unambiguous object descriptions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11–20, 2016.

S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta. R3m: A universal visual representation for robot manipulation. *arXiv preprint arXiv:2203.12601*, 2022.

S. Nasiriany, S. Kirmani, T. Ding, L. Smith, Y. Zhu, D. Driess, D. Sadigh, and T. Xiao. Rt-affordance: Affordances are versatile intermediate representations for robot manipulation. *arXiv preprint arXiv:2411.02704*, 2024.

Octo Model Team, D. Ghosh, H. Walke, K. Pertsch, K. Black, O. Mees, S. Dasari, J. Hejna, C. Xu, J. Luo, T. Kreiman, Y. Tan, P. Sanketi, Q. Vuong, T. Xiao, D. Sadigh, C. Finn, and S. Levine. Octo: An open-source generalist robot policy. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, 2024.

M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

K. Pertsch, K. Stachowicz, B. Ichter, D. Driess, S. Nair, Q. Vuong, O. Mees, C. Finn, and S. Levine. Fast: Efficient action tokenization for vision-language-action models. *arXiv preprint arXiv:2501.09747*, 2025.

Z. Qi, W. Zhang, Y. Ding, R. Dong, X. Yu, J. Li, L. Xu, B. Li, X. He, G. Fan, J. Zhang, J. He, J. Gu, X. Jin, K. Ma, Z. Zhang, H. Wang, and L. Yi. Sofar: Language-grounded orientation bridges spatial reasoning and object manipulation. *CoRR*, abs/2502.13143, 2025. doi: 10.48550/ARXIV.2502.13143. URL https://doi.org/10.48550/arXiv.2502.13143.

D. Qu, H. Song, Q. Chen, Y. Yao, X. Ye, Y. Ding, Z. Wang, J. Gu, B. Zhao, D. Wang, et al. Spatialvla: Exploring spatial representations for visual-language-action model. *arXiv preprint arXiv:2501.15830*, 2025.

A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

K. Rana, J. Haviland, S. Garg, J. Abou-Chakra, I. Reid, and N. Suenderhauf. Sayplan: Grounding large language models using 3d scene graphs for scalable robot task planning. *arXiv preprint arXiv:2307.06135*, 2023.

L. X. Shi, B. Ichter, M. Equi, L. Ke, K. Pertsch, Q. Vuong, J. Tanner, A. Walling, H. Wang, N. Fusai, et al. Hi robot: Open-ended instruction following with hierarchical vision-language-action models. *arXiv preprint arXiv:2502.19417*, 2025.

M. Shukor, D. Aubakirova, F. Capuano, P. Kooijmans, S. Palma, A. Zouitine, M. Aractingi, C. Pascal, M. Russi, A. Marafioti, et al. Smolvla: A vision-language-action model for affordable and efficient robotics. *arXiv preprint arXiv:2506.01844*, 2025.

S. Singh, A. Yadav, J. Jain, H. Shi, J. Johnson, and K. Desai. Benchmarking object detectors with coco: A new path forward. In *European Conference on Computer Vision*, pages 279–295. Springer, 2024.

H. Song, D. Qu, Y. Yao, Q. Chen, Q. Lv, Y. Tang, M. Shi, G. Ren, M. Yao, B. Zhao, et al. Hume: Introducing system-2 thinking in visual-language-action model. *arXiv preprint arXiv:2505.21432*, 2025.

B. R. Team, M. Cao, H. Tan, Y. Ji, M. Lin, Z. Li, Z. Cao, P. Wang, E. Zhou, Y. Han, et al. Robobrain 2.0 technical report. *arXiv preprint arXiv:2507.02029*, 2025.

A. Ten Pas and R. Platt. Using geometry to detect grasp poses in 3d point clouds. In *Robotics Research: Volume 1*, pages 307–324. Springer, 2017.

Y. Tian, S. Yang, J. Zeng, P. Wang, D. Lin, H. Dong, and J. Pang. Predictive inverse dynamics models are scalable learners for robotic manipulation. *arXiv preprint arXiv:2412.15109*, 2024.

W. Wang, Y. Ren, H. Luo, T. Li, C. Yan, Z. Chen, W. Wang, Q. Li, L. Lu, X. Zhu, et al. The all-seeing project v2: Towards general relation comprehension of the open world. In *European Conference on Computer Vision*, pages 471–490. Springer, 2024.

Y. Wang, X. Li, W. Wang, J. Zhang, Y. Li, Y. Chen, X. Wang, and Z. Zhang. Unified vision-language-action model. *arXiv preprint arXiv:2506.19850*, 2025.

J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

K. Wu, C. Hou, J. Liu, Z. Che, X. Ju, Z. Yang, M. Li, Y. Zhao, Z. Xu, G. Yang, et al. Robomind: Benchmark on multi-embodiment intelligence normative data for robot manipulation. *arXiv preprint arXiv:2412.13877*, 2024.

Y. Xie, Z. Xu, M. S. Kankanhalli, K. S. Meel, and H. Soh. Embedding symbolic knowledge into deep networks. *Advances in neural information processing systems*, 32, 2019.

R. Xu, J. Zhang, M. Guo, Y. Wen, H. Yang, M. Lin, J. Huang, Z. Li, K. Zhang, L. Wang, Y. Kuang, M. Cao, F. Zheng, and X. Liang. A0: An affordance-aware hierarchical model for general robotic manipulation, 2025a. URL https://arxiv.org/abs/2504.12636.

R. Xu, J. Zhang, M. Guo, Y. Wen, H. Yang, M. Lin, J. Huang, Z. Li, K. Zhang, L. Wang, et al. A0: An affordance-aware hierarchical model for general robotic manipulation. *arXiv preprint arXiv:2504.12636*, 2025b.

J. Yang, R. Tan, Q. Wu, R. Zheng, B. Peng, Y. Liang, Y. Gu, M. Cai, S. Ye, J. Jang, et al. Magma: A foundation model for multimodal ai agents. *arXiv preprint arXiv:2502.13130*, 2025a.

S. Yang, H. Li, Y. Chen, B. Wang, Y. Tian, T. Wang, H. Wang, F. Zhao, Y. Liao, and J. Pang. Instructvla: Vision-language-action instruction tuning from understanding to manipulation. *arXiv preprint arXiv:2507.17520*, 2025b.

S. Ye, J. Jang, B. Jeon, S. Joo, J. Yang, B. Peng, A. Mandlekar, R. Tan, Y.-W. Chao, B. Y. Lin, L. Liden, K. Lee, J. Gao, L. Zettlemoyer, D. Fox, and M. Seo. Latent action pretraining from videos. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025.

L. Yu, P. Poirson, S. Yang, A. C. Berg, and T. L. Berg. Modeling context in referring expressions. In *European conference on computer vision*, pages 69–85. Springer, 2016.

W. Yuan, J. Duan, V. Blukis, W. Pumacay, R. Krishna, A. Murali, A. Mousavian, and D. Fox. Robopoint: A vision-language model for spatial affordance prediction for robotics. *arXiv preprint arXiv:2406.10721*, 2024.

M. Zawalski, W. Chen, K. Pertsch, O. Mees, C. Finn, and S. Levine. Robotic control via embodied chain-of-thought reasoning. *arXiv preprint arXiv:2407.08693*, 2024.

X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11975–11986, 2023.

Q. Zhao, Y. Lu, M. J. Kim, Z. Fu, Z. Zhang, Y. Wu, Z. Li, Q. Ma, S. Han, C. Finn, et al. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 1702–1713, 2025.

E. Zhou, J. An, C. Chi, Y. Han, S. Rong, C. Zhang, P. Wang, Z. Wang, T. Huang, L. Sheng, et al. Roborefer: Towards spatial referring with reasoning in vision-language models for robotics. *arXiv preprint arXiv:2506.04308*, 2025a.

Z. Zhou, Y. Zhu, J. Wen, C. Shen, and Y. Xu. Vision-language-action model with open-world embodied reasoning from pretrained knowledge. *arXiv preprint arXiv:2505.21906*, 2025b.

J. Zhu, W. Wang, Z. Chen, Z. Liu, S. Ye, L. Gu, H. Tian, Y. Duan, W. Su, J. Shao, Z. Gao, E. Cui, X. Wang, Y. Cao, Y. Liu, X. Wei, H. Zhang, H. Wang, W. Xu, H. Li, J. Wang, N. Deng, S. Li, Y. He, T. Jiang, J. Luo, Y. Wang, C. He, B. Shi, X. Zhang, W. Shao, J. He, Y. Xiong, W. Qu, P. Sun, P. Jiao, H. Lv, L. Wu, K. Zhang, H. Deng, J. Ge, K. Chen, L. Wang, M. Dou, L. Lu, X. Zhu, T. Lu, D. Lin, Y. Qiao, J. Dai, and W. Wang. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models, 2025. URL https://arxiv.org/abs/2504.10479.

A. Author contributions

All contributors are listed in **alphabetical** order by their last names.

A.1. Core Contributors

Yilun Chen, Ning Gao, Jiangmiao Pang, Bolun Wang, Fangjing Wang, Jinhui Ye, Junqiu Yu, Jinyu Zhang, Yangkun Zhu

A.2. Contributors

Xinyi Chen, Weiyang Jin, Hao Li, Yu Qiao, Yang Tian, Bin Wang, Hanqing Wang, Tai Wang, Ziqin Wang, Xueyuan Wei, Chao Wu, Shuai Yang, Jia Zeng, Jingjing Zhang, Shi Zhang, Bowen Zhou