LiLo-VLA: Compositional Long-Horizon Manipulation via Linked Object-Centric Policies (Appendix)

Overview

This supplementary material provides comprehensive details supporting the main paper. Section I presents qualitative visualizations of our real-world experiments, highlighting specific mechanisms for robustness and failure recovery. Section II defines the full atomic skill library and details the exact task sequences for the LIBERO-Long++ and Ultra-Long benchmarks. Section III elaborates on the relative pose representation designed to ensure zero-shot generalization to novel workspace configurations. Section IV offers a theoretical analysis of the combinatorial complexity involved in long-horizon tasks, demonstrating the data efficiency of our modular approach. Section V outlines the training objectives, hyperparameters, and the success checker heuristics utilized in our implementation. Finally, Section VI describes the hardware specifications and perception models utilized in our real-world system.

I. Additional Experimental Results

A. Real-World Task Specifications

To evaluate the compositional generalization of LiLo-VLA, we define a library of 8 unique atomic skills across three distinct scenes. Table I defines these primitives, while Table II details the exact execution sequences for all 8 evaluation tasks.

TABLE I
REAL-WORLD ATOMIC SKILL LIBRARY

Label	Object-Centric Instruction
S_1	"pick the tape"
S_2	"hang the tape"
S_3	"pick the red cup"
S_4	"place the red cup on the plate"
S_5	"pick the yellow mustard"
S_6	"place the yellow mustard in the basket"
S_7	"pick the corn"
S_8	"place the corn in the basket"

B. Real-World Rollout Visualizations

We provide comprehensive visualizations of our real-world experiments in Fig. 1. The top panel displays successful execution rollouts for all 8 evaluation tasks, where we explicitly mark the active object of interest in green and all distractors in red. Note that for any specific atomic skill, there is only one target object; consequently, all other objects in the scene, including targets for other skills, are treated as distractors, highlighting the robustness of our object-centric policy against clutter.

TABLE II
DETAILED TASK SPECIFICATIONS FOR REAL-WORLD

Task ID	Type	Steps	Skill Sequence
Task 1	Standard	4	$S_1 \to S_2 \to S_3 \to S_4$
Task 2	Standard	4	$S_3 \to S_4 \to S_5 \to S_6$
Task 3	Standard	8	$S_7 \to S_8 \to S_5 \to S_6 \to S_3 \to S_4 \to S_1 \to S_2$
Task 4	Clutter	4	$S_1 \to S_2 \to S_3 \to S_4$
Task 5	Clutter	4	$S_3 \to S_4 \to S_5 \to S_6$
Task 6	Permuted	4	$S_3 \to S_4 \to S_1 \to S_2$
Task 7	Permuted	4	$S_5 \to S_6 \to S_3 \to S_4$
Task 8	Permuted	8	$S_3 \to S_4 \to S_1 \to S_2 \to S_7 \to S_8 \to S_5 \to S_6$

To explain LiLo-VLA's robustness, we visualize two core mechanisms. First, for perception, Fig. 2 shows the wrist camera inputs. We apply random black masks to these images, which forces the policy to ignore background clutter and focus only on the target object. Second, for execution, Fig. 3 shows the recovery process. After grasp failures in Frames 1 and 4, the system re-estimates the object's new pose and uses the Reaching Module to reset the arm to the approach pose (Frames 3 and 6). This allows the Interaction Module to retry the skill, eventually leading to a success (Frame 7).

II. Benchmark Details

A. Atomic Skill Library

We define a comprehensive library of 22 atomic skills used across LIBERO-Long++ and Ultra-Long suites. To facilitate concise task specification, we assign a unique label to each skill in Table III.

B. Detailed Task Specifications

We provide the exact atomic skill sequences for all evaluation tasks in Table IV. All skill IDs (e.g., S_1, S_{12}) correspond to the definitions in the Atomic Skill Library (Table III).

For Suite 1 (LIBERO-Long++), we define 6 core tasks focused on robustness and reordering. Each task consists of a Standard sequence and a corresponding Variant, where the latter permutes the execution order of atomic skills to evaluate the policy's zero-shot compositional generalization.

For Suite 2 (Ultra-Long), we define 3 complex long-horizon scenarios to evaluate extreme temporal scalability. Each scenario includes one Standard sequence and two additional

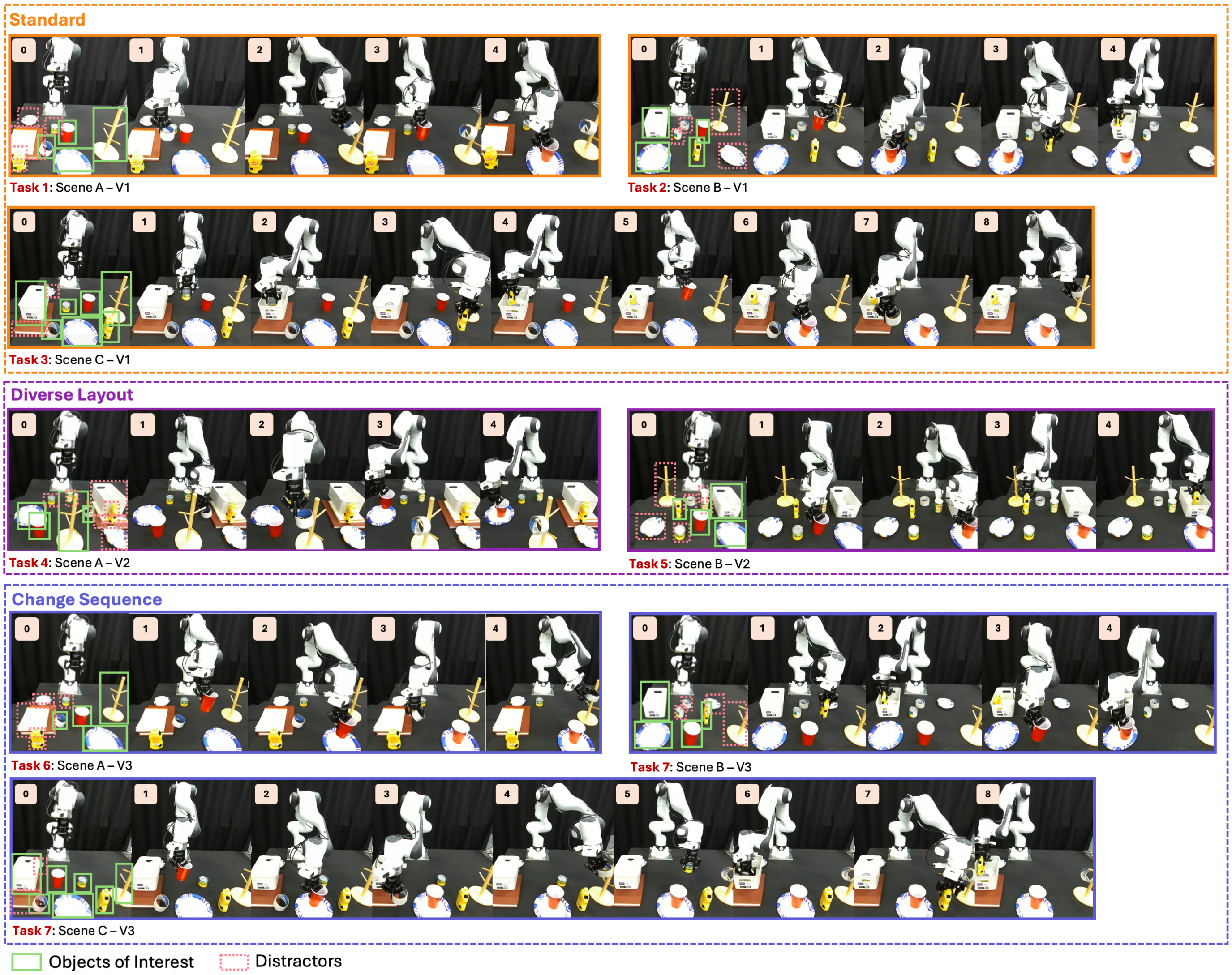

Fig. 1. **Qualitative Real-World Experimental Results.**

TABLE III
SIMULATION ATOMIC SKILL LIBRARY (LIBERO-LONG++ AND ULTRA-LONG)

ID	Skill Description	ID	Skill Description
S_1	Pick Alphabet Soup	S_{12}	Pick Moka Pot
S_2	Place Alphabet Soup in Basket	S_{13}	Place Moka Pot on Stove
S_3	Pick Cream Cheese	S_{14}	Turn On Stove
S_4	Place Cream Cheese in Basket	S_{15}	Pick Butter
S_5	Pick Tomato Sauce	S_{16}	Place Butter in Basket
S_6	Place Tomato Sauce in Basket	S_{17}	Pick Chocolate Pudding
S_7	Pick Black Bowl	S_{18}	Place Chocolate Pudding Right of Plate
S_8	Place Black Bowl on Plate	S_{19}	Pick White Mug
S_9	Stack Black Bowl on Black Bowl	S_{20}	Place White Mug on Plate
S_{10}	Place Black Bowl in Bottom Drawer	S_{21}	Pick Yellow and White Mug
S_{11}	Close Bottom Drawer	S_{22}	Place Yellow and White Mug on Right Plate

Variant sequences. Notably, the “Table Organization” task extends up to 16 steps, serving as a rigorous stress test for the system’s ability to maintain coherent execution over extended horizons.

III. EXTENDED METHODOLOGY DETAILS

While the main text details our object-centric visual processing, the handling of proprioception is equally critical for generalization. To further enforce an object-centric inductive bias in our simulation backbone (OpenVLA-OFT), we transform the standard absolute end-effector pose into a relative frame. Specifically, the policy input is computed as $T^{ee}_{obj} = (T^{obj}_{world})^{-1} T^{ee}_{world}$, where T^{ee}_{world} and T^{obj}_{world} represent the global poses of the end-effector and target object, respectively. This formulation renders the interaction policy invariant to global workspace shifts. In contrast, our real-world Pi0.5 policy operates directly in joint space and thus does not utilize this Cartesian transformation. We empirically validate the robustness of this design by introducing random spatial perturbations of up to 20cm (where the object’s position

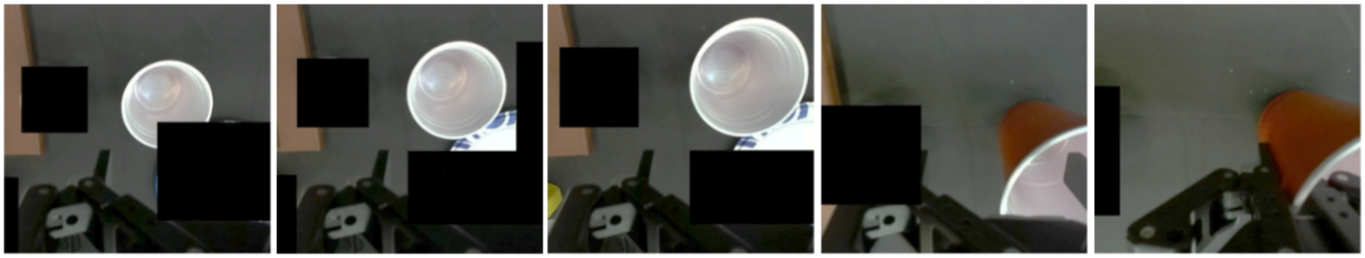

Fig. 2. **Visual Robustness.** Random masking on wrist camera inputs forces the policy to focus strictly on the target object effectively ignoring background clutter.

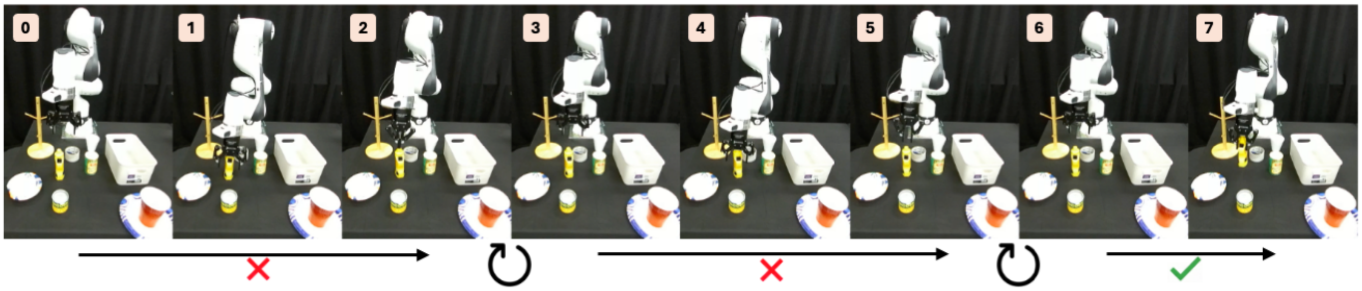

Fig. 3. **Recovery Mechanism.** Upon detecting a grasp failure, the system autonomously re-estimates the object pose and resets the arm to retry the execution.

is randomly placed within this range); the system maintains strong performance with no statistically significant degradation (overall success rate over all 22 atomic skills in III dropped only slightly from 0.80 to 0.75, $p > 0.05$).

IV. Data Efficiency and Combinatorial Complexity

To rigorously feature the data efficiency advantage of LiLo-VLA over end-to-end baselines, we analyze the sample complexity required to generalize to novel long-horizon task sequences. Let a long-horizon task $\mathcal{T}$ be composed of a set of atomic skills $\mathcal{S} = \{s_1, s_2, \ldots, s_N\}$. In a Task and Motion Planning (TAMP) formulation, the valid execution orders are governed by causal dependencies, where the effect of a skill s_i serves as a precondition for a subsequent skill s_j. These dependencies induce a partial ordering over $\mathcal{S}$, denoted as $\prec$.

End-to-End Complexity. An end-to-end policy π_{e2e} must implicitly internalize these valid orderings from demonstration data. To achieve robust compositional generalization, the policy must observe sufficient coverage of the set of all valid linearizations (i.e., topological sorts) consistent with the partial order $\prec$. For a task involving M independent objects, each requiring a pick-place sequence, the number of valid linearizations can be as large as $\Theta\left(\frac{(2M)!}{2^M}\right)$ when objects have no inter-dependencies. For our ultra-long tasks ($N = 16$, $M = 8$), this combinatorial explosion renders it intractable to cover the distribution of valid trajectories via demonstrations alone.

Modular Complexity (LiLo-VLA). In contrast, LiLo-VLA decouples the task into independent atomic execution units. Since our Interaction Module π_{int} is strictly object-centric and conditioned only on the immediate target, the learning problem reduces to mastering the set of unique atomic skills $\mathcal{U} \subseteq \mathcal{S}$. Consequently, the data requirement scales linearly, $\mathcal{O}(|\mathcal{U}|)$, independent of the total horizon length N or the combinatorial complexity of the task structure. For the 16-step "Table Organization" task, this reduces the learning problem from covering millions of potential trajectory variations to simply mastering the 9 unique atomic primitives listed in Table I.

V. Training and Implementation Details

A. *Training Objectives*

We employ two distinct training objectives corresponding to the different backbones used in our simulation and real-world experiments.

OpenVLA-OFT (Simulation). For our simulation benchmarks, we utilize OpenVLA-OFT [2], which departs from the standard discrete tokenization used in original VLA models. Instead, it adopts a *continuous action representation* and employs an Optimized Fine-Tuning (OFT) recipe that integrates parallel decoding and action chunking. Consequently, the model is trained via a regression objective rather than classification. We minimize the L1 loss between the predicted

TABLE IV
DETAILED TASK SPECIFICATIONS FOR SIMULATION BENCHMARKS

Suite	Task Name	Steps	Skill Sequence (IDs from Table III)
Suite 1: LIBERO-Long++ (Robustness and Reordering)			
Standard	T1: Stove and Moka	3	$S_{14} \rightarrow S_{12} \rightarrow S_{13}$
Variant	T2: Stove and Moka	3	$S_{12} \rightarrow S_{13} \rightarrow S_{14}$
Standard	T3: Soup and Cheese	4	$S_1 \rightarrow S_2 \rightarrow S_3 \rightarrow S_4$
Variant	T4: Soup and Cheese	4	$S_3 \rightarrow S_4 \rightarrow S_1 \rightarrow S_2$
Standard	T5: Soup and Tomato	4	$S_1 \rightarrow S_2 \rightarrow S_5 \rightarrow S_6$
Variant	T6: Soup and Tomato	4	$S_5 \rightarrow S_6 \rightarrow S_1 \rightarrow S_2$
Standard	T7: Cheese and Butter	4	$S_3 \rightarrow S_4 \rightarrow S_{15} \rightarrow S_{16}$
Variant	T8: Cheese and Butter	4	$S_{15} \rightarrow S_{16} \rightarrow S_3 \rightarrow S_4$
Standard	T9: Two Mugs	4	$S_{19} \rightarrow S_{20} \rightarrow S_{21} \rightarrow S_{22}$
Variant	T10: Two Mugs	4	$S_{21} \rightarrow S_{22} \rightarrow S_{19} \rightarrow S_{20}$
Standard	T11: Mug and Pudding	4	$S_{19} \rightarrow S_{20} \rightarrow S_{17} \rightarrow S_{18}$
Variant	T12: Mug and Pudding	4	$S_{17} \rightarrow S_{18} \rightarrow S_{19} \rightarrow S_{20}$
Suite 2: Ultra-Long (Scalability and Compositionality)			
Standard	T13: Kitchen Organization	9	$S_7 \rightarrow S_8 \rightarrow S_7 \rightarrow S_8 \rightarrow S_7 \rightarrow S_9 \rightarrow S_3 \rightarrow S_4 \rightarrow S_{11}$
Variant 1	T14: Kitchen Org. (V2)	9	$S_3 \rightarrow S_4 \rightarrow S_7 \rightarrow S_8 \rightarrow S_7 \rightarrow S_9 \rightarrow S_7 \rightarrow S_8 \rightarrow S_{11}$
Variant 2	T15: Kitchen Org. (V3)	9	$S_7 \rightarrow S_8 \rightarrow S_7 \rightarrow S_9 \rightarrow S_3 \rightarrow S_4 \rightarrow S_7 \rightarrow S_8 \rightarrow S_{11}$
Standard	T16: Cooking Preparation	10	$S_{14} \rightarrow S_{12} \rightarrow S_{13} \rightarrow S_{14} \rightarrow S_{12} \rightarrow S_{13} \rightarrow S_7 \rightarrow S_8 \rightarrow S_7 \rightarrow S_8$
Variant 1	T17: Cooking Prep. (V2)	10	$S_7 \rightarrow S_8 \rightarrow S_7 \rightarrow S_8 \rightarrow S_{14} \rightarrow S_{12} \rightarrow S_{13} \rightarrow S_{14} \rightarrow S_{12} \rightarrow S_{13}$
Variant 2	T18: Cooking Prep. (V3)	10	$S_7 \rightarrow S_8 \rightarrow S_{14} \rightarrow S_{12} \rightarrow S_{13} \rightarrow S_7 \rightarrow S_8 \rightarrow S_{14} \rightarrow S_{12} \rightarrow S_{13}$
Standard	T19: Table Organization	16	$S_5 \rightarrow S_6 \rightarrow S_5 \rightarrow S_6 \rightarrow S_1 \rightarrow S_2 \rightarrow S_1 \rightarrow S_2 \rightarrow S_3 \rightarrow S_4 \rightarrow S_3 \rightarrow S_4 \rightarrow S_7 \rightarrow S_{10} \rightarrow S_7 \rightarrow S_8$
Variant 1	T20: Table Org. (V2)	16	$S_1 \rightarrow S_2 \rightarrow S_5 \rightarrow S_6 \rightarrow S_1 \rightarrow S_2 \rightarrow S_5 \rightarrow S_6 \rightarrow S_3 \rightarrow S_4 \rightarrow S_3 \rightarrow S_4 \rightarrow S_7 \rightarrow S_{10} \rightarrow S_7 \rightarrow S_8$
Variant 2	T21: Table Org. (V3)	16	$S_1 \rightarrow S_2 \rightarrow S_1 \rightarrow S_2 \rightarrow S_5 \rightarrow S_6 \rightarrow S_5 \rightarrow S_6 \rightarrow S_3 \rightarrow S_4 \rightarrow S_3 \rightarrow S_4 \rightarrow S_7 \rightarrow S_{10} \rightarrow S_7 \rightarrow S_8$

action chunk $\hat{\mathbf{a}}_{t:t+H}$ and the ground-truth action sequence $\mathbf{a}_{t:t+H}$:

$$\mathcal{L}_{OFT} = \frac{1}{H} \sum_{k=0}^{H-1} \| \hat{\mathbf{a}}_{t+k} - \mathbf{a}_{t+k} \|_1 \tag{1}$$

where H is the action chunk size, and the model predicts the entire chunk in parallel conditioned on the observation o_t and instruction l.

Pi0.5 (Real-World). For real-world validation, we adopt the Pi0.5 backbone [1], which models the continuous action distribution using Conditional Flow Matching (CFM). The model learns a vector field v_θ that transports a Gaussian noise distribution $x_0 \sim \mathcal{N}(0, I)$ to the data distribution x_1 (ground-truth actions) over a virtual time $\tau \in [0, 1]$. The training objective minimizes the mean squared error between the predicted vector field and the target flow:

$$\mathcal{L}_{FM} = \mathbb{E}_{\tau, x_0, x_1} \left[|| v_\theta(\phi_\tau(x_0, x_1), \tau, o_t, l) - (x_1 - x_0) ||^2 \right] \tag{2}$$

where $\phi_\tau(x_0, x_1) = (1 - \tau)x_0 + \tau x_1$ represents the linear interpolation path.

B. Training Hyperparameters

We train a single multi-task policy for each domain: one OpenVLA-OFT model across the simulation atomic skill library and one Pi0.5 model across the real-world atomic skill library. Table V provides a comprehensive overview of the hyperparameters used for these trainings.

C. Success Checker Heuristic Functions

In our simulation experiments, the verification function $\mathcal{V}(a_i)$ leverages ground-truth state information to evaluate skill execution. We primarily adopt the standard success predicates provided by the LIBERO benchmark. We create one success condition for the Pick skill: the skill is deemed successful only if the target object's vertical position (z-axis) increases by at least 3 cm relative to its pre-execution state. For all other atomic skills, we retain the default LIBERO success conditions without modification.

TABLE V
HYPERPARAMETERS AND TRAINING CONFIGURATION.

Hyperparameter	**Simulation (OpenVLA-OFT)**	**Real-World (Pi0.5)**
Base Model	OpenVLA-7B	Pi0.5 (DROID)
Action Representation	Continuous (OFT)	Continuous (Flow Matching)
Finetuning Method	LoRA	Full
Action Chunk Size (H)	8	16
Input Observation	Wrist RGB + Proprioception	Wrist RGB + Proprioception
Training Objective	L1 Regression	Flow Matching
Optimizer	AdamW	AdamW
Learning Rate	5×10^{-4}	5×10^{-5}
LR Schedule	Cosine Decay	Cosine Decay
Batch Size	64	32
Training Steps	100,000	30,000
Random Erasing	Enabled	Enabled

VI. HARDWARE AND SYSTEM SETUP

A. Hardware Specifications

Our real-world experimental setup follows the DROID hardware configuration standards. We use a 7-DoF Franka Emika Panda robotic arm equipped with a standard parallel jaw gripper (Robotiq 2F-85). The robot is operated using a joint impedance controller at a control frequency of 15 Hz. The vision system comprises two stereo cameras: a wrist-mounted Stereolabs ZED Mini for local object-centric observations and a single fixed third-person Stereolabs ZED 2 camera for global reaching. The VLA policy inference runs on a remote cluster node equipped with NVIDIA RTX A6000 GPUs, while all perception modules and real-time control loops operate on a

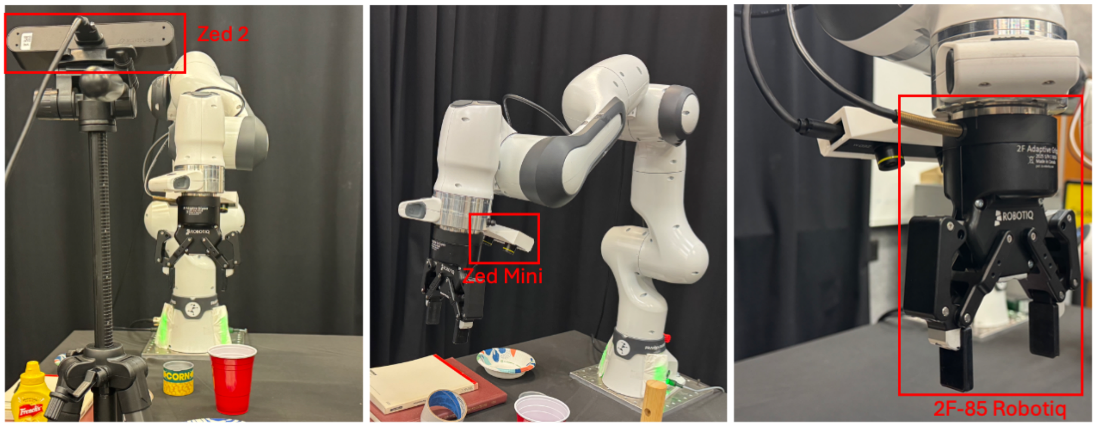

Fig. 4. **Real-world hardware setup**. Our system consists of a Franka Emika Panda arm, a wrist-mounted ZED mini camera for fine-grained interaction, and a fixed ZED 2i camera for global reaching. Compute is distributed between a local workstation (Perception/Control) and a remote cluster (VLA Inference).

local workstation equipped with a single NVIDIA RTX 4080 GPU. We visualize the complete physical setup in Fig. 4.

B. *Perception Models*

We utilize FoundationPose [4] for 6D pose estimation and YOLOE [3] for 2D object detection and segmentation. Both models operate in real-time to enable closed-loop feedback. FoundationPose tracks the target object pose using an RGB image, an instance segmentation mask, and a 3D CAD model. We obtain the necessary 3D meshes by scanning the physical objects with the AR Code mobile application. For object detection, we employ YOLOE with visual prompting. Instead of conditioning detection on natural language text descriptions, we provide a reference image with a bounding box of the target object. We find this visual prompting approach offers higher robustness against background clutter compared to open-vocabulary text prompts.

References

[1] Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. π0. 5: a vision-language-action model with open-world generalization, 2025. *URL https://arxiv. org/abs/2504.16054*, 1 (2):3.

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

[3] Ao Wang, Lihao Liu, Hui Chen, Zijia Lin, Jungong Han, and Guiguang Ding. Yoloe: Real-time seeing anything, 2025. URL https://arxiv.org/abs/2503.07465.

[4] Bowen Wen, Wei Yang, Jan Kautz, and Stan Birchfield. Foundationpose: Unified 6d pose estimation and tracking of novel objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17868–17879, 2024.