MiVLA: Towards Generalizable Vision-Language-Action Model with Human-Robot Mutual Imitation Pre-training

Supplementary Material

A Simulation Experiment Details

This section aims to provide more comprehensive settings and results for the simulation experiments mentioned in the main text. We will detail the domain randomization parameters employed in the "Hard" mode of the RoboTwin-2.0 benchmark, and present the complete evaluation results across all 50 tasks for a more thorough analysis.

A.1 Domain Randomization Settings

To evaluate the robustness and generalization capabilities of our model, we introduce domain randomization in the "Hard" mode by perturbing key properties of the environment to simulate real-world diversity. Based on our experimental setup, we primarily adopt the following three randomization strategies:

- **Visual Background and Texture Randomization:** The background of the simulated environment is randomly selected and applied from a diverse texture library.
- **Table Clutter Randomization:** Distractor objects of various geometric shapes and colors are randomly placed in non-critical areas of the workspace to increase scene complexity and visual clutter.
- **Lighting Condition Randomization:** The position, color, and intensity of the scene's light sources are sampled within a predefined range to simulate different lighting conditions.

In addition, we introduce minor geometric perturbations to the environment, such as random variations in the workbench height within a range of ± 3 cm.

Figure A1. Comparison of the 'easy mode' (top-left) and 'hard mode' (remaining images) environments in RoboTwin-2.0.

A.2 Complete Simulation Evaluation Results

For the sake of brevity in the main text, Section 4.2 reported the evaluation results on a representative subset of 20 tasks. To provide a more comprehensive performance landscape, this section presents the success rates (SR) of all baseline models across the entire suite of 50 tasks in the RoboTwin-2.0 benchmark.Figures A2 and A3 showcase examples of MiVLA's successful trajectories in representative tasks.The complete evaluation results are presented in Table A1.These data further corroborate the conclusions drawn in the main text: our proposed MiVLA not only excels in the representative subset of tasks but also maintains a comprehensive performance lead across the entire task suite.

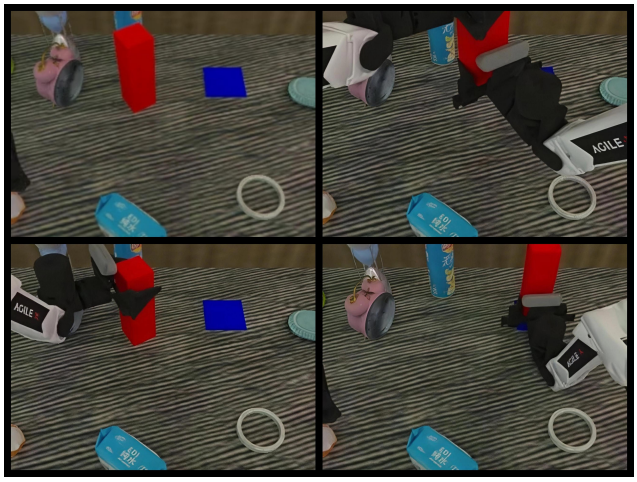

Figure A2. An example of MiVLA's performance in the handover_block task within RoboTwin

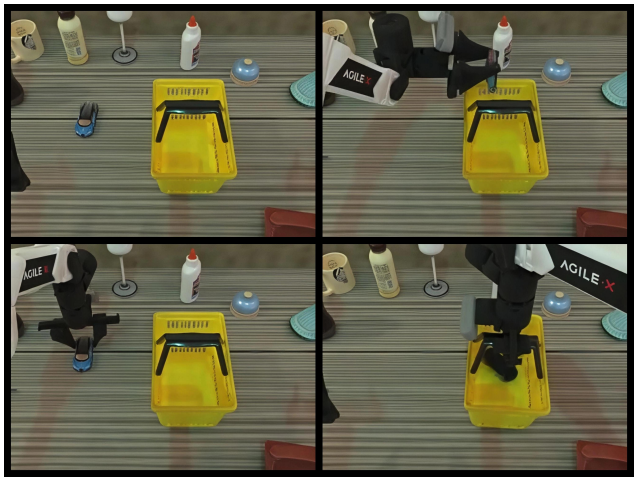

Figure A3. An example of MiVLA's performance in the "place_object_basket" task within RoboTwin

B. Real-World Robot Experiment Details

B.1 Robot Embodiment Specifications

In this study, we employed three distinct robot embodiments with significant differences to rigorously evaluate the model's cross-embodiment generalization capabilities.

- **AgileX PiPer & ARX-5:** Both are table-top 6-DoF single-arm manipulators. Although they share the same number of degrees of freedom, they exhibit significant differences in their joint ranges, dynamic properties, and rotational joint characteristics.
- **LocoMan:** A composite robot embodiment composed of a quadruped robot and a lightweight dual-arm system. Its unique 6-DoF manipulation capability is provided by a hybrid combination of two parts: (1)The first three DoF are realized through the leg movements of the quadruped; (2)The last three DoF are provided by three Dynamixel servos mounted on the robot's front legs. This hybrid-driven kinematic structure serves as a challenging test case to evaluate whether our VLA model can generalize knowledge learned from standard robot morphologies and adapt to a novel embodiment with a disparate structure.

B.2 Data Collection

To fine-tune the model, 30 successful expert demonstration trajectories were collected for every real-world task using two teleoperation methods. The AgileX PiPER and ARX-5 were controlled via a Leader-Follower scheme, where demonstrations were generated by an operator manipulating a kinematically identical master arm. The LocoMan was operated using a human pose-based solution, in which an operator's head and hand poses, tracked by an Apple Vision Pro, were mapped in real-time to the robot's base locomotion and dual-arm commands.

B.3 Qualitative Results and Analysis

To supplement the quantitative indicators presented in the experimental section of the text, this section provides a qualitative visualization of the final results of policy implementation in several representative practical tasks. Figure **??** shows the comparison of the performance of MiVLA and all benchmark models in several representative scenarios. These visual results offer a deeper insight into the behavior of the model. From the observations, it can be seen that the baseline methods often exhibit some common failure modes, such as inaccurate grasping, item dropping, or inability to reach the target state. In contrast, The MiVLA model demonstrates higher accuracy and stronger stability, and is able to successfully complete tasks in all different robot forms. This not only highlights its effectiveness in terms of success rate, but also reflects the quality and time of the executed trajectories.

C. Limitations and Future Work

Despite the strong performance of our MiVLA model, it is important to acknowledge its limitations, which primarily surface in out-of-distribution (OOD) scenarios. We identify three representative failure modes when the model encounters novel objects, unseen initial poses, and distracting backgrounds

- **Novel Objects:** The model may struggle to generate appropriate grasping poses for objects with shapes and textures significantly different from the training data (Figure A4a).
- **Unseen Initial Poses:** When objects are placed in highly unusual or cluttered initial positions, the policy sometimes fails to find a valid trajectory, leading to collisions or inaction (Figure A4b).
- **Distracting Backgrounds:** Although trained with domain randomization, the model can still be distracted by highly complex or visually salient backgrounds that were not well-represented in the training distribution, causing it to misinterpret the task goal (Figure A4c).

(a) Failure on a new object

(b) Failure on unseen pose

(c) Failure on a distracting object

Figure A4. An example of MiVLA's performance in the "handover_block" task within RoboTwin

It is important to note that while our MiVLA model demonstrates considerable generalization capabilities, its limitations become apparent in these more extreme or difficult out-of-distribution scenarios. We observe that the $\pi_{0.5}$ baseline exhibits better semantic generalization in some of these challenging cases. This superior performance can be attributed to its foundational architecture: $\pi_{0.5}$ [4] is built upon a Vision-Language Model (VLM) that was pretrained on a large-scale, multi-source, and heterogeneous dataset.

This highlights a fundamental trade-off between different architectural designs. Our MiVLA, based on a Diffusion Transformer, excels at directly learning complex visuomotor policies from demonstration data. However, it lacks the explicit semantic and commonsense reasoning capabilities that VLMs acquire through their extensive pretraining. In essence, while our model masters the *how* of a task through visual pattern recognition, models like $\pi_{0.5}$, owing to their VLM foundation, possess a better understanding of the what and why.

To address these limitations, a promising future direction is to integrate the cognitive reasoning abilities of VLMs with the powerful generative capabilities of diffusion-based policies. By leveraging a pretrained VLM to provide semantic guidance—such as object grounding, affordance prediction, or high-level planning—this fusion has the potential to enable the model to handle abstract language instructions, recover from errors through reasoning, and ultimately lead to more generalizable and human-like robotic intelligence.

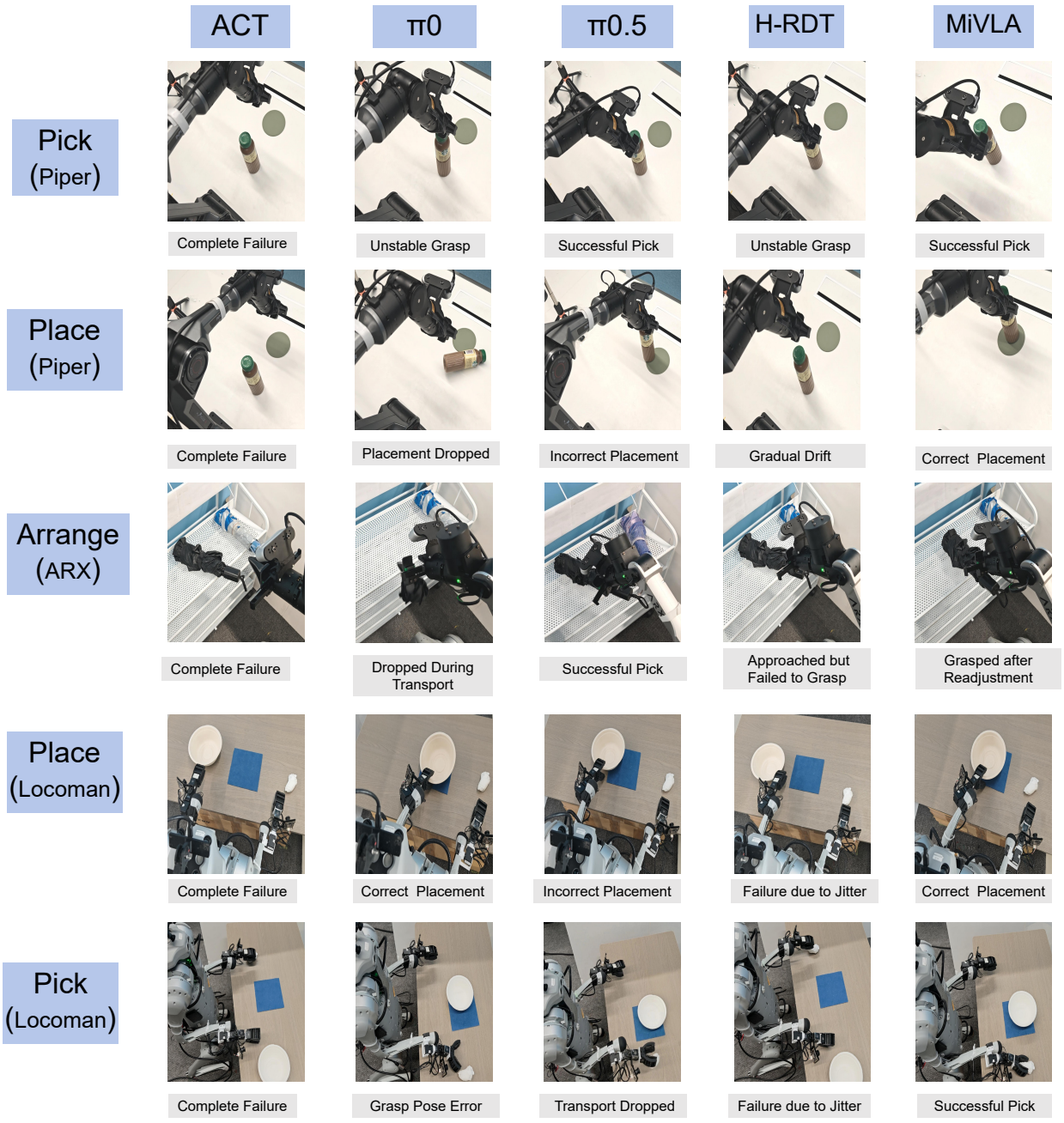

Figure A5. Qualitative comparison of policy performance on various real-world tasks and robot embodiments.

	ACT		Π_0		$\Pi_{0.5}$		H-RDT		MiVLA	
Task Name	Easy	Hard	Easy	Hard	Easy	Hard	Easy	Hard	Easy	Hard
blocks_ranking_rgb	0%	0%	1%	5%	**17%**	42%	3%	2%	13%	**47%**
blocks_ranking_size	0%	0%	0%	1%	3%	17%	3%	3%	**21%**	**34%**
handover_block	0%	1%	1%	2%	12%	22%	3%	3%	**66%**	**42%**
hanging_mug	0%	0%	5%	3%	6%	14%	5%	5%	**19%**	**25%**
move_can_pot	0%	0%	21%	18%	42%	50%	48%	34%	**74%**	**68%**
move_stapler_pad	0%	0%	2%	5%	13%	26%	4%	8%	**30%**	**33%**
place_a2b_left	1%	0%	4%	4%	15%	49%	16%	25%	**51%**	**55%**
place_object_basket	0%	1%	29%	40%	22%	57%	7%	40%	**71%**	**74%**
stack_blocks_two	0%	1%	17%	18%	**33%**	**68%**	2%	2%	23%	7%
stack_bowls_three	0%	0%	22%	28%	54%	62%	46%	60%	**79%**	**70%**
put_bottles_dustbin	0%	0%	7%	1%	13%	8%	9%	2%	**47%**	**33%**
put_object_cabinet	0%	0%	0%	0%	0%	1%	0%	0%	**20%**	**15%**
press_stapler	22%	21%	67%	60%	70%	71%	57%	65%	**78%**	**85%**
open_microwave	7%	1%	7%	12%	64%	66%	74%	64%	**76%**	**79%**
move_playingcard_away	2%	0%	30%	42%	33%	84%	20%	49%	**76%**	79%
open_laptop	6%	3%	33%	35%	80%	96%	60%	78%	**98%**	**99%**
dump_bin_bigbin	5%	16%	33%	49%	54%	82%	95%	81%	**98%**	**99%**
handover_mic	10%	10%	16%	37%	45%	89%	71%	94%	**98%**	**99%**
grab_roller	33%	60%	60%	73%	63%	99%	69%	80%	**100%**	**100%**
click_bell	51%	22%	62%	55%	20%	28%	75%	83%	**100%**	**99%**
click_alarmclock	36%	22%	53%	50%	57%	62%	61%	74%	**100%**	**100%**
adjust_bottle	1%	10%	45%	69%	25%	97%	57%	90%	**100%**	**96%**
beat_block_hammer	0%	5%	44%	35%	69%	64%	25%	35%	**95%**	**83%**
lift_pot	0%	16%	5%	8%	54%	84%	27%	31%	**100%**	**95%**
blocks_ranking_rgb	0%	0%	1%	5%	**17%**	**42%**	0%	0%	0%	2%
move_pillbottle_pad	0%	1%	5%	7%	13%	43%	8%	26%	**70%**	**71%**
pick_diverse_bottles	3%	0%	14%	10%	31%	44%	17%	20%	**59%**	**63%**
pick_dual_bottles	5%	2%	13%	17%	40%	34%	4%	26%	**58%**	**57%**
place_a2b_right	1%	0%	2%	6%	25%	41%	10%	28%	**62%**	**73%**
place_bread_basket	1%	0%	10%	14%	9%	43%	5%	29%	**56%**	**53%**
place_bread_skillet	2%	0%	8%	2%	16%	42%	8%	12%	**62%**	**51%**
place_burger_fries	7%	11%	20%	12%	15%	68%	12%	44%	**79%**	**83%**
place_can_basket	1%	1%	4%	6%	11%	40%	14%	31%	**41%**	**59%**
place_cans_plasticbox	1%	2%	0%	9%	0%	40%	16%	33%	**39%**	**56%**
place_dual_shoes	1%	0%	6%	8%	17%	34%	1%	7%	**28%**	**36%**
place_empty_cup	3%	4%	17%	32%	57%	87%	29%	67%	**90%**	**88%**
place_fan	0%	0%	6%	3%	29%	56%	16%	29%	**75%**	**75%**
place_mouse_pad	0%	0%	2%	4%	3%	16%	0%	10%	**24%**	**28%**
place_object_scale	0%	0%	7%	7%	16%	56%	8%	21%	**44%**	**63%**
place_object_stand	0%	1%	25%	25%	56%	75%	22%	42%	**61%**	**76%**
place_phone_stand	2%	0%	6%	5%	31%	53%	13%	19%	**65%**	**71%**
place_shoe	0%	0%	19%	30%	35%	68%	18%	31%	**89%**	**83%**
rotate_qrcode	2%	0%	5%	17%	52%	66%	39%	71%	**84%**	**86%**
scan_object	0%	0%	**1%**	0%	**1%**	0%	0%	0%	0%	0%
shake_bottle	27%	20%	89%	73%	93%	**98%**	87%	84%	**99%**	**98%**
shake_bottle_horizontally	26%	19%	95%	82%	91%	**97%**	85%	86%	**99%**	**97%**
stack_blocks_three	0%	0%	1%	2%	12%	34%	0%	0%	1%	0%
stack_bowls_two	11%	7%	14%	74%	87%	**97%**	85%	94%	**91%**	93%
stamp_seal	0%	0%	11%	16%	19%	**43%**	4%	10%	**29%**	31%
turn_switch	1%	6%	9%	19%	38%	35%	34%	31%	**62%**	**71%**
Average(All 50 tasks)	5.4%	5.4%	19.1%	21.7%	33.6%	53.8%	27.4%	37.2%	**62.0%**	**63.6%**

Table A1. Complete success rates of all methods on the 50 tasks of the RoboTwin-2.0 benchmark. Results are reported for both "Easy" and "Hard" variations. The best-performing method, MiVLA, is frequently highlighted in bold.