Beyond Imitation: Reinforcement Learning–Based Sim–Real Co-Training for VLA Models

Anonymous Submission

***Abstract*—Simulation offers a scalable and low-cost way to enrich vision-language-action (VLA) training, reducing reliance on expensive real-robot demonstrations. However, most sim-real co-training methods rely on supervised fine-tuning (SFT), which treats simulation as a static source of demonstrations and does not exploit large-scale closed-loop interaction. Consequently, real-world gains and generalization are often limited. In this paper, we propose an *RL*-based sim-real *Co*-training (RL-Co) framework that leverages interactive simulation while preserving real-world capabilities. Our method follows a generic two-stage design: we first warm-start the policy with SFT on a mixture of real and simulated demonstrations, then fine-tune it with reinforcement learning in simulation while adding an auxiliary supervised loss on real-world data to anchor the policy and mitigate catastrophic forgetting. We evaluate our framework on four real-world tabletop manipulation tasks using two representative VLA architectures, OpenVLA and $\pi_{0.5}$, and observe consistent improvements over real-only fine-tuning and SFT-based co-training, including +24% real-world success on OpenVLA and +20% on $\pi_{0.5}$. Beyond higher success rates, RL co-training yields stronger generalization to unseen task variations and substantially improved real-world data efficiency, providing a practical and scalable pathway for leveraging simulation to enhance real-robot deployment.**

I. Introduction

Building general-purpose robots that reliably solve real-world tasks remains a central challenge in robotics. Vision-language–action (VLA) models have recently emerged as a promising paradigm, achieving strong results in robotic manipulation [1–6] and visual navigation [7–13]. These models are typically pretrained on large-scale real-world demonstrations [14–16], learning perception and control from expert data. However, despite extensive pretraining, performance often degrades under novel scenes and task variations [17]. Moreover, collecting large-scale real-robot data remains costly and time-consuming.

Simulation provides a scalable alternative. Modern simulators [18–22] and large open-source asset libraries [23–26] enable diverse training environments at scale. Early approaches relied on domain randomization [27–29] to bridge the sim-to-real gap, but required careful manual design and struggled with complex manipulation. More recent real-to-sim-to-real pipelines [30–34] and generative modeling approaches [35–38] improve visual fidelity and diversity. Nevertheless, high-fidelity simulation demands accurate modeling of geometry, materials, contacts, and sensing, increasing system complexity and limiting scalability across tasks.

Beyond direct sim-to-real transfer, recent work [6, 30, 31, 39–44] explores sim–real co-training that jointly leverages simulated and real data. By leveraging scalable simulation data, these methods outperform real-only training and remain effective even under visual mismatch [41] or loosely related tasks [42]. However, most existing approaches remain purely supervised, treating simulation as static demonstration data and failing to exploit scalable interactive learning.

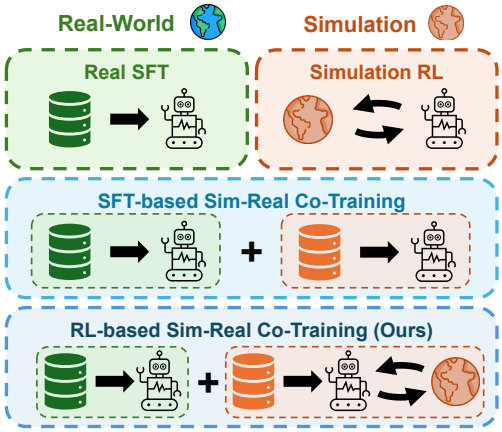

Fig. 1: **Overview of training paradigms combining real-world and simulated data.** VLA models are commonly trained via supervised fine-tuning (SFT) on real-world demonstrations, via RL in simulation followed by sim-to-real transfer, or via SFT-based sim–real co-training. We instead propose an RL-based sim–real co-training (RL-Co) framework, which initializes with sim–real SFT and then performs RL in simulation with real-world SFT regularization.

Meanwhile, behavior cloning via SFT is prone to compounding errors under distribution shift [45]. Recent work [46–50] investigates reinforcement learning (RL) as a post-training paradigm for VLA models, achieving higher success rates and stronger generalization in simulation. Yet these methods typically rely on zero-shot sim-to-real transfer with domain randomization, often resulting in substantial performance drops on real robots.

In this work, we propose an *RL*-based sim-real *Co*-training (RL-Co) framework that combines interactive simulation with real-world supervision. Our two-stage approach first initializes the policy via supervised co-training on mixed real and simulated demonstrations, then performs RL in simulation while incorporating an auxiliary supervised loss on real-world data to prevent catastrophic forgetting and preserve real-world capabilities.

We validate RL-Co on four real-world tabletop manipulation tasks using two representative VLA models, OpenVLA [5] and $\pi_{0.5}$ [4]. Across tasks and models, RL-Co consistently

outperforms real-only fine-tuning and SFT-based sim–real co-training, achieving higher real-world success rates. In addition, our method demonstrates stronger generalization to unseen variations, greater robustness to hyperparameter choices, and substantially improved real-world data efficiency, providing a scalable pathway for deploying VLA models on real robots.

II. Related Works

A. Vision-Language-Action Models for Manipulation Tasks

Vision-Language-Action (VLA) models integrate visual perception and linguistic reasoning into a unified foundation model for robotic control [1–5, 51]. Built on advances in Large Language Models and Vision-Language Models [52–56], they are typically pretrained on large-scale internet images [57–59] and robotic demonstrations [14, 60]. Such pretraining enables strong generalization, allowing VLAs to follow natural language instructions and perform diverse manipulation tasks across embodiments.

B. Fine-Tuning VLA Models via Reinforcement Learning

Post-training adapts pretrained VLA models to downstream manipulation tasks. Most methods rely on Supervised Fine-Tuning (SFT), which aligns policies with target distributions using limited demonstrations [61–63]. However, SFT suffers from covariate shift, where compounding errors cause policies to deviate from expert trajectories [45, 46, 64].

Recent work introduces reinforcement learning (RL) into post-training, enabling improvement through interaction and trial-and-error. Depending on the VLA architecture, diverse RL strategies have been explored [46, 47, 49, 50, 65–68]. For example, [49] exploit temperature sampling in OpenVLA [5] for PPO-based fine-tuning [69], while [47] introduce stochasticity into flow matching denoising [70] to improve exploration. However, most RL-based VLA training remains simulation-based for safety and efficiency, requiring sim-to-real transfer or extensive domain randomization. Direct real-world RL avoids this gap [65–68, 71] but is constrained by cost, safety risks, and slow data collection [72]. In contrast, our method bridges simulated RL and real-world data constraints to enable efficient policy improvement without heavy sim-to-real engineering.

C. Sim-to-Real Transfer and Sim-Real Co-Training

Simulation provides scalable and safe robotic learning, but the sim-to-real gap remains a core challenge. High-fidelity digital twins reduce this gap through accurate visual and physical modeling [73–76], yet are costly and still struggle to capture real-world complexity. Domain Randomization (DR) instead improves robustness by randomizing visual and physical parameters during simulation [27–29, 74, 77], but often requires extensive training and careful tuning to avoid overly conservative policies.

Recent work shifts toward sim-real co-training, jointly optimizing policies with simulated and real-world data [6, 30, 31, 39–44]. Some approaches reduce the domain gap by learning invariant representations shared across simulation and reality [39, 40, 78], while others treat simulation as large-scale data augmentation to improve generalization even with limited visual fidelity or task alignment [21, 30, 31, 42, 79, 80].

However, most co-training methods treat simulation as a static trajectory source, overlooking its interactive nature. Our method builds on data augmentation while incorporating reinforcement learning into the co-training loop, enabling active exploration in simulation and grounding policies with real-world data.

III. Preliminaries

A. Problem Formulation

For each real-world manipulation task T_{real}, we construct a corresponding digital-twin simulation environment, resulting in a simulation task T_{sim} [81]. The simulator mirrors the real setup while enabling scalable data collection through interaction. Both real and simulated tasks are modeled as Partially Observable Markov Decision Processes (POMDPs):

$$\mathcal{M}_\Omega = \langle \mathcal{S}_\Omega, \mathcal{A}, \mathcal{P}_\Omega, \mathcal{R}, \mathcal{O}_\Omega, \mathcal{L}, P(s_0), \gamma \rangle, \tag{1}$$

where $\Omega \in \{\text{real}, \text{sim}\}$ denotes the real or simulated environment.

Following [42], we define each component as follows: $\mathcal{S}_\Omega$ and $\mathcal{O}_\Omega$ denote the system state and observation spaces, which differ across environments but share the same robot embodiment and sensing modalities. Both domains use an identical action space $\mathcal{A}$ and control interface. State transitions follow $s_{t+1} \sim \mathcal{P}_\Omega(\cdot \mid s_t, a_t)$, where simulation dynamics may differ slightly from real-world physics. The language instruction $\mathcal{L}$ specifies the task goal and is shared across domains. The reward function $\mathcal{R}(s, l)$ evaluates task progress given the current state and instruction. Both tasks share the same initial state distribution $s_0 \sim P(s_0)$, and $\gamma \in (0, 1)$ denotes the discount factor.

Under this formulation, a vision-language-action (VLA) policy π_θ conditions on the most recent H observations $o_\Omega^{t-H+1:t}$ and instruction l to predict a sequence of future actions $a_{t:t+h-1}$.

B. Fine-Tuning on Vision-Language-Action Models

We study post-training of vision-language-action (VLA) policies under supervised and reinforcement learning paradigms. Given a pretrained policy π_θ, fine-tuning adapts the model to downstream manipulation tasks using expert demonstrations or interactive environment feedback.

1) Supervised Fine-Tuning (SFT): Given an expert demonstration dataset $\mathcal{D}_T = \{(\tau^{(i)}, l^{(i)})\}_{i=1}^N$, each trajectory $\tau^{(i)} = \{(o_j^{(i)}, a_j^{(i)})\}_{j=1}^{K_i}$ contains observation–action pairs with instruction $l^{(i)}$. SFT trains π_θ by minimizing the discrepancy between predicted and expert actions:

$$L_{\text{SFT}}(\theta) = \mathop{\mathbb{E}}_{(\tau,l)\sim D_T,\ t\sim\text{Unif}(\{1\sim K_\tau\})} \Big[\ell_{\text{SFT}}(\hat{a}_{t:t+h-1}, a_{t:t+h-1}) \Big], \tag{2}$$

where the predicted action chunk is $\hat{a}_{t:t+h-1} = \pi_\theta(o_{t-H+1:t}, l)$, and $a_{t:t+h-1}$ denotes the corresponding expert actions.

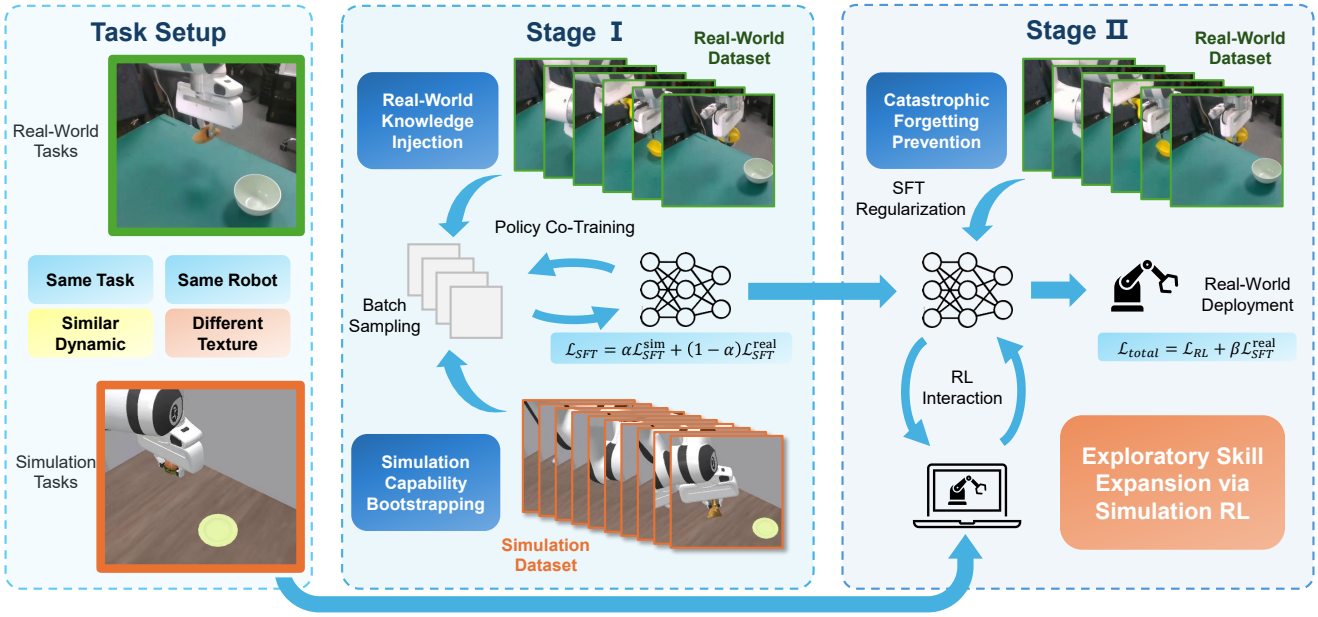

Fig. 2: **Overview of the proposed two-stage sim-real co-training framework.** We construct a digital-twin setup where T_{sim} corresponds to T_{real} despite visual discrepancies. **Stage I** initializes the policy via SFT on mixed real and simulated data (ratio α). **Stage II** performs RL in simulation while applying a real-world SFT loss as regularization to preserve real-world capabilities.

2) Reinforcement Learning (RL) Fine-Tuning: RL fine-tuning further improves the policy through interaction by maximizing the expected discounted return:

$$\pi^* = \arg\max_{\pi_\theta} \mathbb{E}_{\pi_\theta,\mathcal{P}}\Big[\sum_{t=0}^{\infty}\gamma^t\mathcal{R}(s_t,l)\Big], \tag{3}$$

where actions follow $a_t \sim \pi_\theta(\cdot|o_t,l)$ and transitions follow $s_{t+1}\sim\mathcal{P}(s_t,a_t)$.

Although RL implementations vary across VLA architectures, existing methods generally alternate between trajectory collection and reward-guided policy updates. Our method follows this framework and introduces an additional supervised objective on real-world data during RL optimization, making it compatible with a wide range of RL fine-tuning algorithms.

C. SFT-based Co-Training

Given a real-world task T_{real} and its digital-twin simulation counterpart T_{sim}, we assume access to expert demonstrations $\mathcal{D}_{\text{real}}$ and $\mathcal{D}_{\text{sim}}$ collected in the respective environments. A straightforward way to leverage both sources is supervised co-training, which jointly fine-tunes the VLA policy using mixed real and simulated data. The objective is defined as a weighted combination of SFT losses:

$$\mathcal{L}_{\text{SFT}}(\theta) = \alpha\,\mathcal{L}_{\text{SFT}}(\theta;\mathcal{D}_{\text{sim}}) + (1-\alpha)\,\mathcal{L}_{\text{SFT}}(\theta;\mathcal{D}_{\text{real}}), \tag{4}$$

where $\alpha\in[0,1]$ controls the proportion of simulated supervision. Following [42], this is equivalently implemented by sampling trajectories from $\mathcal{D}_{\text{sim}}$ with probability α and from $\mathcal{D}_{\text{real}}$ with probability $1-\alpha$.

Although widely adopted for sim-to-real transfer, SFT-based co-training remains limited by imitation learning, as it depends on demonstration quality and cannot exploit reward feedback or online interaction. These limitations motivate our RL-based co-training approach.

IV. Method

In this section, we present our *RL*-based sim-real *Co*-training (RL-Co) framework, illustrated in Fig. 2.

A. Stage I: SFT Co-Training for Policy Initialization

Starting from a pre-trained VLA policy π_θ not adapted to the target tasks, Stage I initializes the policy using both real-world and simulated demonstrations. We perform supervised fine-tuning co-training on the real-world dataset $\mathcal{D}_{\text{real}}$ and the simulation dataset $\mathcal{D}_{\text{sim}}$, as described in Section III-C.

This stage serves two purposes. First, it incorporates task-specific real-world knowledge required for deployment. Second, learning from simulated demonstrations establishes sufficient competence in simulation, ensuring a non-trivial success rate and providing a suitable initialization for reinforcement learning. These properties motivate SFT co-training as the first stage of our framework. A detailed analysis is provided in Section V-C.

B. Stage II: Sim-Real Co-Training with Real-Regularized RL

While Stage I equips the policy with real-world and simulated capabilities, its optimization remains limited to imitation objectives. Stage II expands policy competence through online interaction in simulation while preventing degradation of real-world performance.

We introduce an auxiliary supervised fine-tuning objective on real-world data into the reinforcement learning process. During simulation RL training, policy updates are driven

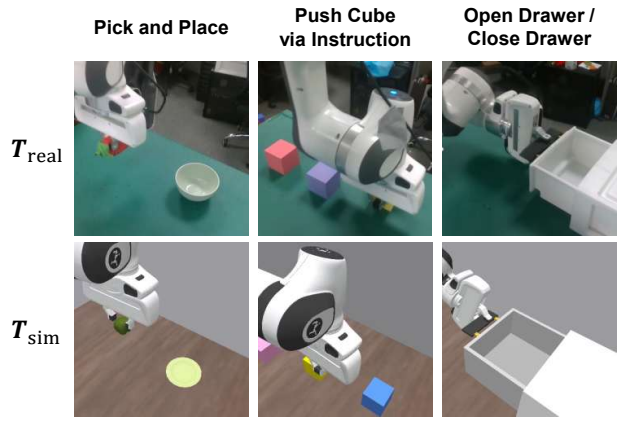

Fig. 3: **Visualization of our tabletop manipulation tasks.** The top row shows third-person real-world observations, while the bottom row presents the corresponding simulated views.

by the reinforcement learning loss $\mathcal{L}_{\text{RL}}$, which promotes exploration and reward maximization. We augment it with an additional SFT loss computed on $\mathcal{D}_{\text{real}}$, yielding:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{RL}} + \beta\, \mathcal{L}_{\text{SFT}}(\theta; \mathcal{D}_{\text{real}}), \tag{5}$$

where β balances reinforcement learning updates and preservation of real-world knowledge.

The RL term leverages large-scale simulated interaction to explore diverse behaviors and improve performance, while real-world supervision regularizes the policy toward demonstrations, mitigating catastrophic forgetting during RL fine-tuning. This simple modification is compatible with a wide range of RL fine-tuning algorithms and forms the core of our RL-Co framework.

V. Experiments

In this section, we empirically evaluate the proposed RL-Co framework and aim to answer the following questions:

- Does RL-Co improve real-world performance compared to training with real-world data only or SFT-based sim–real co-training?
- How do the individual components in our two-stage framework contribute to the final performance?
- To what extent can our method reduce the amount of required real-world demonstration data?

A. Experimental Setting

1) Environmental Setting: To evaluate the proposed model, we design four tabletop manipulation tasks requiring diverse perception, language grounding, and control capabilities. An overview of the real-world and simulated environments is shown in Fig. 3.

- `Pick and Place`. Grasp objects with varying shapes and place them into a target container.
- `Push Cube via Instruction`. Push the correct cube according to a natural language instruction.
- `Open Drawer`. Open a closed drawer on the table.
- `Close Drawer`. Close an opened drawer on the table.

Simulation environments are built using ManiSkill [20], matching the real-world camera viewpoints and scene layout. Rather than photorealistic reconstruction, we model only essential object geometry required for task execution.

Across all tasks, the camera pose and robot initialization are fixed, while object positions are randomly sampled within predefined regions. All methods are evaluated on identical initial states for fair comparison. Each setting is evaluated twice and reported as task success rate.

For each task, we collect 20–50 real-world expert demonstrations via teleoperation, and generate 1,000 successful trajectories in simulation for training.

2) Implementation: To validate the generality of RL-Co, we implement our method on two representative VLA policies: OpenVLA [5] and the flow-matching-based $\pi_{0.5}$ model [4]. In the SFT stage, real-world and simulation datasets are directly mixed and used for training with the official open-source training pipelines.

For OpenVLA, we follow [46] and extend the codebase by introducing the real-world regularization loss during RL optimization. For $\pi_{0.5}$, we adopt ReinFlow [47] as the RL algorithm. RL training is conducted using RLinf [82], with the real-world regularization term integrated into the RL objective.

For all RL experiments, we fix the total number of environment interaction steps for each model–task pair and train in simulation until convergence.

B. Main Results

To evaluate the effectiveness of RL-Co, we compare our method with two baselines: real-world-only SFT and SFT-based sim–real co-training. Quantitative results are reported in Table I.

Fine-tuning VLA models using only limited real-world demonstrations results in poor performance across most tasks. This limitation is particularly evident for OpenVLA, whose success rates remain below 20% in all environments. The $\pi_{0.5}$ model performs better on the simpler `Pick and Place` task, but still struggles in more challenging scenarios. Introducing simulation data through SFT-based co-training improves performance on easier tasks (e.g., `Close Drawer`) but provides limited gains on harder ones. Moreover, when real-only SFT already performs well, co-training can occasionally degrade performance, indicating that imitation-based sim–real mixing does not reliably translate simulated data into real-world improvements.

In contrast, RL-Co consistently achieves substantially higher real-world success rates across all model–task combinations, with three settings improving by more than 35%. These results demonstrate that reinforcement learning enables simulated interaction to enhance policy capability more effectively than both baselines.

Improvement of Generalization by RL-Co. To further analyze the role of reinforcement learning, we evaluate generalization under distribution shifts. Using the `Pick and Place` task with the $\pi_{0.5}$ policy, we consider two unseen settings: (i) *Unseen Objects*, involving novel object categories,

VLA Model	Experiment Setting	Pick and Place	Push Cube	Open Drawer	Close Drawer	Avg
OpenVLA	Real-Only Training	6.3 ± 0.0	20.0 ± 13.3	0.0 ± 0.0	10.0 ± 10.0	16.5 ± 13.3
	SFT Co-Training	23.4 ± 4.7	51.7 ± 5.0	0.0 ± 0.0	85.0 ± 5.0	40.0 ± 3.7
	RL-Co (Ours)	**58.8** ± 10.0	**68.3** ± 11.7	**35.0** ± 15.0	**95.0** ± 5.0	**64.0** ± 0.7
$\pi_{0.5}$	Real-Only Training	71.9 ± 9.4	0.0 ± 0.0	0.0 ± 0.0	35.0 ± 15.0	26.7 ± 1.4
	SFT Co-Training	68.8 ± 9.4	10.0 ± 3.3	10.0 ± 0.0	95.0 ± 5.0	45.9 ± 4.4
	RL-Co (Ours)	**81.3** ± 9.4	**18.4** ± 1.7	**65.0** ± 5.0	**100.0** ± 0.0	**66.2** ± 4.0

TABLE I: **Comparison of real-world success rates under different training paradigms.** We compare our RL-Co approach with real-only SFT and SFT co-training across four tabletop manipulation tasks, evaluated on both OpenVLA and $\pi_{0.5}$. Results are reported in terms of success rate (SR, %). All values are presented as mean ± standard deviation.

Experiment Setting	In-Distribution	Unseen Objects	Unseen States
Real-Only	71.9	25.0 (46.9↓)	40.0 (31.95↓)
SFT Co-Training	68.8	31.3 (37.5↓)	55.0 (13.8↓)
RL-Co (Ours)	**81.3**	**56.3** (25.0↓)	**70.0** (11.3↓)

TABLE II: **Comparison of generalization under unseen settings.** We evaluate all $\pi_{0.5}$ models on the Pick and Place task under out-of-distribution conditions, including unseen objects and unseen states. We report the success rate (SR, %) as well as the relative performance drop compared to the in-distribution setting.

and (ii) *Unseen States*, where the robot initial pose is perturbed. Results are summarized in Table II.

Under the original setting, all methods achieve comparable performance, with RL-Co slightly outperforming the baselines. However, real-only SFT degrades sharply under distribution shifts, dropping by more than 45% for unseen objects and 30% for unseen states, indicating limited robustness. SFT-based co-training improves generalization, showing higher success rates in both unseen settings, but still suffers substantial degradation, especially for unseen objects (over 35% drop).

In contrast, RL-Co demonstrates significantly stronger generalization, with markedly smaller performance drops in both unseen-object and unseen-state evaluations. These results suggest that RL enables policies to acquire more robust and transferable behaviors beyond supervised co-training alone.

Impact of Different SFT Co-Training Ratios α and Real-World Regularization Weights β. We further study two key hyperparameters: the data mixture ratio α used during SFT co-training, and the real-world regularization weight β applied during RL fine-tuning. Experiments are conducted on the Pick and Place and Open Drawer tasks using the $\pi_{0.5}$ model. We first vary α during SFT co-training, then select one model for RL co-training with different β values.

As shown in Fig. 4, the mixture ratio α strongly affects SFT co-training performance. For Pick and Place, where real-only training already performs well, increasing simulated data degrades real-world performance. In contrast, for the more challenging Open Drawer task, both very small and excessively large simulation ratios perform poorly, and intermediate α values provide a better balance between supervision sources.

The regularization weight β also significantly influences performance. However, across all evaluated β values, RL co-training consistently produces large improvements over the corresponding SFT models, achieving higher success rates

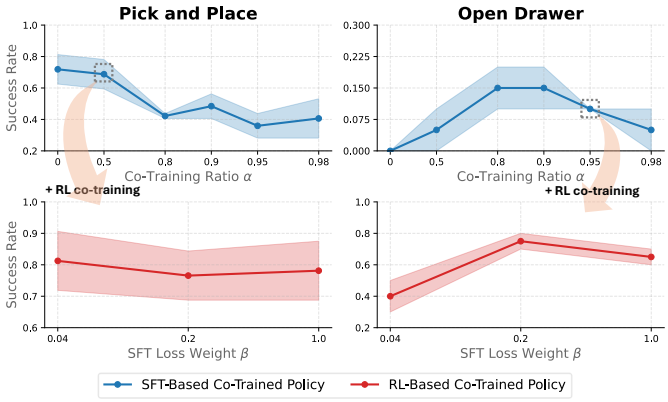

Fig. 4: **Analysis of the co-training ratio (α) and regularization weight (β).** We vary α for SFT co-training on Pick and Place and Open Drawer. RL co-training is evaluated under different β values using $\alpha = 0.5$ for Pick and Place and $\alpha = 0.95$ for Open Drawer. Shaded regions denote standard deviation.

than all SFT-only settings across different α. These results show that reinforcement learning effectively extends the performance boundary of SFT-based co-training.

C. Ablation Study

We conduct ablation studies to analyze the contribution of each component in RL-Co, focusing on two questions: (i) how simulation data in Stage I influences RL optimization, and (ii) the role of real-world supervision in Stage I and Stage II.

1) Effect of Simulation Data in Stage I: To evaluate the importance of simulated data in Stage I, we directly perform RL co-training from a policy trained only on real-world demonstrations. As shown in Fig. 5, this initialization leads to extremely poor sample efficiency, with near-trivial simulation success rates even after more than three million interaction steps. In contrast, SFT co-training with simulated demonstrations provides a much stronger initialization and enables efficient RL optimization. These results demonstrate that simulation data in Stage I is critical for effective RL-based co-training.

2) Role of Real-World Supervision in Two Stages: We further study the role of real-world supervision by removing it from Stage I and Stage II separately. Fig. 6 reports real-world success rates on the Pick and Place task using the $\pi_{0.5}$ model.

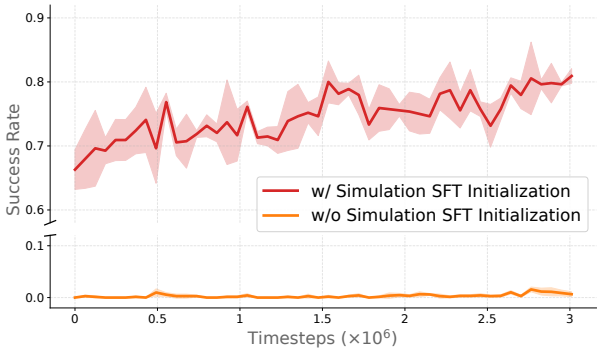

Fig. 5: **Ablation study on simulation SFT initialization.** Simulation success rate during RL training with and without simulation SFT initialization. Results are averaged over three random seeds, with shaded regions indicating standard deviation.

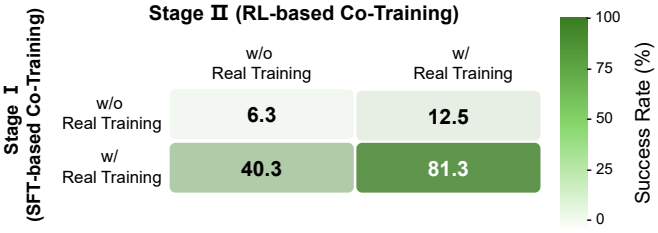

Fig. 6: **Ablation study on real-world supervision.** We ablate real-world supervised training in Stage I and Stage II separately and report the resulting real-world success rates.

Removing real-world SFT regularization from Stage II reduces the success rate from 81.38% to 40.25%, indicating catastrophic forgetting during RL optimization in simulation despite improved simulated performance. Removing real-world SFT from Stage I further degrades performance to 12.5%, highlighting the higher data efficiency of SFT compared with RL when leveraging limited real-world demonstrations. Since RL relies on extensive simulator interaction, the real-world SFT term in Stage II mainly acts as a regularizer to preserve learned real-world skills. Finally, removing real-world supervision from both stages results in a collapse to 6.25%, showing that zero-shot transfer from low-fidelity simulation alone remains highly challenging.

D. Data Efficiency

As shown in Section V-B, RL-Co outperforms both real-only training and SFT-based co-training under the same amount of real-world supervision. We further study its data efficiency by evaluating how much real-world data can be reduced compared to these baselines. We conduct experiments on `Open Drawer` task. The real-world dataset is expanded to 200 expert demonstrations, and we evaluate real-only SFT, SFT co-training, and RL-Co under varying amounts of real-world data. Results are reported in Fig. 7.

As, expected, all methods benefit from additional demonstrations, showing steady performance improvements as data increases. With simulated data, SFT co-training improves

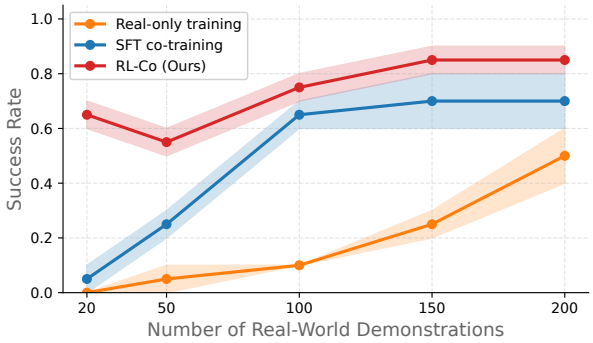

Fig. 7: **Effect of the number of real-world demonstrations.** Performance on the `Open Drawer` task using the $\pi_{0.5}$ model under different amounts of real-world data. Shaded regions denote standard deviation.

faster than real-only training, reaching 65% success with 100 demonstrations, already exceeding real-only training using all 200 demonstrations. However, both baselines remain clearly inferior to RL-Co: even with 200 demonstrations, their performance is comparable to or worse than our method trained with only 20 demonstrations. These results highlight the substantial real-world data efficiency of RL-Co under the evaluated settings.

VI. Conclusion

This paper presents RL-Co, an RL-based sim–real co-training framework for vision-language-action (VLA) models that overcomes limitations of prior methods relying mainly on supervised fine-tuning. RL-Co follows a two-stage pipeline compatible with diverse learning algorithms and VLA architectures. The policy is first initialized via supervised fine-tuning on mixed simulated and real demonstrations, and then optimized with reinforcement learning in simulation while an auxiliary supervised loss on real-world data preserves real-world behaviors. By leveraging online interaction and reward feedback, RL-Co moves beyond static imitation, reduces compounding errors, and mitigates catastrophic forgetting common in supervised training or simulation-only RL.

Extensive real-world experiments across tasks and popular VLA models validate our approach. RL-Co consistently outperforms real-only fine-tuning and SFT-based co-training, achieving higher real-world success rates, stronger robustness, and improved data efficiency. These results highlight the potential of reinforcement learning to better leverage simulation in co-training and surpass imitation-only objectives.

Limitations. Despite promising results, several limitations remain. We evaluate only tabletop manipulation on a single robot embodiment and do not study heterogeneous sim–real settings. Although RL-Co improves real-world success, performance remains below 100%, and real-world RL is not yet incorporated, which may further enhance robustness. Future work will extend the framework to more diverse tasks, longer-horizon manipulation, additional embodiments, and more efficient sim–real RL co-training with improved sim-to-real alignment.

References

[1] B. Zitkovich, T. Yu, S. Xu, P. Xu, T. Xiao, F. Xia, J. Wu, P. Wohlhart, S. Welker, A. Wahid *et al.*, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," in *Conference on Robot Learning*. PMLR, 2023, pp. 2165–2183.

[2] 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.

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

[4] 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.

[5] 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.

[6] 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.

[7] M. Savva, A. Kadian, O. Maksymets, Y. Zhao, E. Wijmans, B. Jain, J. Straub, J. Liu, V. Koltun, J. Malik *et al.*, "Habitat: A platform for embodied ai research," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 9339–9347.

[8] C. Bizer and A. Schultz, "The r2r framework: Publishing and discovering mappings on the web." *COLD*, vol. 665, pp. 97–108, 2010.

[9] A. Ku, P. Anderson, R. Patel, E. Ie, and J. Baldridge, "Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding," *arXiv preprint arXiv:2010.07954*, 2020.

[10] J. Thomason, M. Murray, M. Cakmak, and L. Zettlemoyer, "Vision-and-dialog navigation," in *Conference on Robot Learning*. PMLR, 2020, pp. 394–406.

[11] Y. Hong, Q. Wu, Y. Qi, C. Rodriguez-Opazo, and S. Gould, "Vln bert: A recurrent vision-and-language bert for navigation," in *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, 2021, pp. 1643–1653.

[12] W. Hao, C. Li, X. Li, L. Carin, and J. Gao, "Towards learning a generic agent for vision-and-language navigation via pre-training," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 13 137–13 146.

[13] P.-L. Guhur, M. Tapaswi, S. Chen, I. Laptev, and C. Schmid, "Airbert: In-domain pretraining for vision-and-language navigation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 1634–1643.

[14] A. O'Neill, A. Rehman, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain *et al.*, "Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 6892–6903.

[15] 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.

[16] H. R. Walke, K. Black, T. Z. Zhao, Q. Vuong, C. Zheng, P. Hansen-Estruch, A. W. He, V. Myers, M. J. Kim, M. Du *et al.*, "Bridgedata v2: A dataset for robot learning at scale," in *Conference on Robot Learning*. PMLR, 2023, pp. 1723–1736.

[17] S. Zhang, Z. Xu, P. Liu, X. Yu, Y. Li, Q. Gao, Z. Fei, Z. Yin, Z. Wu, Y.-G. Jiang, and X. Qiu, "Vlabench: A large-scale benchmark for language-conditioned robotics manipulation with long-horizon reasoning tasks," 2024. [Online]. Available: https://arxiv.org/abs/2412.18194

[18] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2012, pp. 5026–5033.

[19] 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.

[20] S. Tao, F. Xiang, A. Shukla, Y. Qin, X. Hinrichsen, X. Yuan, C. Bao, X. Lin, Y. Liu, T.-k. Chan *et al.*, "Maniskill3: Gpu parallelized robotics simulation and rendering for generalizable embodied ai," *arXiv preprint arXiv:2410.00425*, 2024.

[21] T. Mu, Z. Ling, F. Xiang, D. Yang, X. Li, S. Tao, Z. Huang, Z. Jia, and H. Su, "Maniskill: Generalizable manipulation skill benchmark with large-scale demonstrations," *arXiv preprint arXiv:2107.14483*, 2021.

[22] J. Gu, F. Xiang, X. Li, Z. Ling, X. Liu, T. Mu, Y. Tang, S. Tao, X. Wei, Y. Yao *et al.*, "Maniskill2: A unified benchmark for generalizable manipulation skills," *arXiv preprint arXiv:2302.04659*, 2023.

[23] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su *et al.*, "Shapenet: An information-rich 3d model repository," *arXiv preprint arXiv:1512.03012*, 2015.

[24] M. Deitke, R. Liu, M. Wallingford, H. Ngo, O. Michel, A. Kusupati, A. Fan, C. Laforte, V. Voleti, S. Y. Gadre *et al.*, "Objaverse-xl: A universe of 10m+ 3d objects," *Advances in Neural Information Processing Systems*, vol. 36, pp. 35 799–35 813, 2023.

[25] M. Deitke, D. Schwenk, J. Salvador, L. Weihs, O. Michel, E. VanderBilt, L. Schmidt, K. Ehsani, A. Kembhavi, and A. Farhadi, "Objaverse: A universe of annotated 3d objects," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 13 142–13 153.

[26] B. Calli, A. Singh, A. Walsman, S. Srinivasa, P. Abbeel, and A. M. Dollar, "The ycb object and model set: Towards common benchmarks for manipulation research," in *2015 international conference on advanced robotics (ICAR)*. IEEE, 2015, pp. 510–517.

[27] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2017, pp. 23–30.

[28] O. M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray *et al.*, "Learning dexterous in-hand manipulation," *The International Journal of Robotics Research*, vol. 39, no. 1, pp. 3–20, 2020.

[29] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Sim-to-real transfer of robotic control with dynamics randomization," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 3803–3810.

[30] T. Chen, Z. Chen, B. Chen, Z. Cai, Y. Liu, Z. Li, Q. Liang, X. Lin, Y. Ge, Z. Gu *et al.*, "Robotwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation," *arXiv preprint arXiv:2506.18088*, 2025.

[31] S. Nasiriany, A. Maddukuri, L. Zhang, A. Parikh, A. Lo, A. Joshi, A. Mandlekar, and Y. Zhu, "Robocasa: Large-scale simulation of everyday tasks for generalist robots," *arXiv preprint arXiv:2406.02523*, 2024.

[32] Y. Wu, L. Pan, W. Wu, G. Wang, Y. Miao, F. Xu, and H. Wang, "Rl-gsbridge: 3d gaussian splatting based real2sim2real method for robotic manipulation learning," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025, pp. 192–198.

[33] K. Zhang, S. Sha, H. Jiang, M. Loper, H. Song, G. Cai, Z. Xu, X. Hu, C. Zheng, and Y. Li, "Real-to-sim robot policy evaluation with gaussian splatting simulation of soft-body interactions," *arXiv preprint arXiv:2511.04665*, 2025.

[34] X. Li, J. Li, Z. Zhang, R. Zhang, F. Jia, T. Wang, H. Fan, K.-K. Tseng, and R. Wang, "Robogsim: A real2sim2real robotic gaussian splatting simulator," *arXiv preprint arXiv:2411.11839*, 2024.

[35] Y. Shen, F. Wei, Z. Du, Y. Liang, Y. Lu, J. Yang, N. Zheng, and B. Guo, "Videovla: Video generators can be generalizable robot manipulators," *arXiv preprint arXiv:2512.06963*, 2025.

[36] G. R. Team, K. Choromanski, C. Devin, Y. Du, D. Dwibedi, R. Gao, A. Jindal, T. Kipf, S. Kirmani, I. Leal *et al.*, "Evaluating gemini robotics policies in a veo world simulator," *arXiv preprint arXiv:2512.10675*, 2025.

[37] F. Zhu, H. Wu, S. Guo, Y. Liu, C. Cheang, and T. Kong, "Irasim: Learning interactive real-robot action simulators," *arXiv preprint arXiv:2406.14540*, 2024.

[38] S. Zhou, Y. Du, Y. Yang, L. Han, P. Chen, D.-Y. Yeung, and C. Gan, "Learning 3d persistent embodied world models," *arXiv preprint arXiv:2505.05495*, 2025.

[39] A. Yu, A. Foote, R. Mooney, and R. Martín-Martín, "Natural language can help bridge the sim2real gap," *arXiv preprint arXiv:2405.10020*, 2024.

[40] J. Yang, C. Finn, and D. Sadigh, "Invariance co-training for robot visual generalization," *arXiv preprint arXiv:2512.05230*, 2025.

[41] A. Wei, A. Agarwal, B. Chen, R. Bosworth, N. Pfaff, and R. Tedrake, "Empirical analysis of sim-and-real cotraining of diffusion policies for planar pushing from pixels," *arXiv preprint arXiv:2503.22634*, 2025.

[42] A. Maddukuri, Z. Jiang, L. Y. Chen, S. Nasiriany, Y. Xie, Y. Fang, W. Huang, Z. Wang, Z. Xu, N. Chernyadev *et al.*, "Sim-and-real co-training: A simple recipe for vision-based robotic manipulation," *arXiv preprint arXiv:2503.24361*, 2025.

[43] L. Ankile, A. Simeonov, I. Shenfeld, M. Torne, and P. Agrawal, "From imitation to refinement-residual rl for precise assembly," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025, pp. 01–08.

[44] K. Fang, W. Liang, Y. Li, J. Zhang, P. Zeng, L. Gao, J. Song, and H. T. Shen, "Sim-and-human co-training for data-efficient and generalizable robotic manipulation," 2026. [Online]. Available: https://arxiv.org/abs/2601.19406

[45] S. Ross, G. Gordon, and D. Bagnell, "A reduction of imitation learning and structured prediction to no-regret online learning," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2011, pp. 627–635.

[46] J. Liu, F. Gao, B. Wei, X. Chen, Q. Liao, Y. Wu, C. Yu, and Y. Wang, "What can rl bring to vla generalization? an empirical study," *arXiv preprint arXiv:2505.19789*, 2025.

[47] T. Zhang, C. Yu, S. Su, and Y. Wang, "Reinflow: Fine-tuning flow matching policy with online reinforcement learning," *arXiv preprint arXiv:2505.22094*, 2025.

[48] Y. Zhang, S. Yu, T. Zhang, M. Guang, H. Hui, K. Long, Y. Wang, C. Yu, and W. Ding, "Sac flow: Sample-efficient reinforcement learning of flow-based policies via velocity-reparameterized sequential modeling," *arXiv preprint arXiv:2509.25756*, 2025.

[49] H. Li, Y. Zuo, J. Yu, Y. Zhang, Z. Yang, K. Zhang, X. Zhu, Y. Zhang, T. Chen, G. Cui *et al.*, "Simplevla-rl: Scaling vla training via reinforcement learning," *arXiv preprint arXiv:2509.09674*, 2025.

[50] J. Liu, G. Liu, J. Liang, Y. Li, J. Liu, X. Wang, P. Wan, D. Zhang, and W. Ouyang, "Flow-grpo: Training

flow matching models via online rl," *arXiv preprint arXiv:2505.05470*, 2025.

[51] O. M. Team, D. Ghosh, H. Walke, K. Pertsch, K. Black, O. Mees, S. Dasari, J. Hejna, T. Kreiman, C. Xu *et al.*, "Octo: An open-source generalist robot policy," *arXiv preprint arXiv:2405.12213*, 2024.

[52] A. Yang, A. Li, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Gao, C. Huang, C. Lv *et al.*, "Qwen3 technical report," *arXiv preprint arXiv:2505.09388*, 2025.

[53] G. Team, A. Kamath, J. Ferret, S. Pathak, N. Vieillard, R. Merhej, S. Perrin, T. Matejovicova, A. Ramé, M. Rivière *et al.*, "Gemma 3 technical report," *arXiv preprint arXiv:2503.19786*, 2025.

[54] 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*, 2025.

[55] L. Beyer, A. Steiner, A. S. Pinto, A. Kolesnikov, X. Wang, D. Salz, M. Neumann, I. Alabdulmohsin, M. Tschannen, E. Bugliarello *et al.*, "Paligemma: A versatile 3b vlm for transfer," *arXiv preprint arXiv:2407.07726*, 2024.

[56] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.

[57] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman *et al.*, "Laion-5b: An open large-scale dataset for training next generation image-text models," *Advances in neural information processing systems*, vol. 35, pp. 25 278–25 294, 2022.

[58] S. Changpinyo, P. Sharma, N. Ding, and R. Soricut, "Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts," 2021. [Online]. Available: https://arxiv.org/abs/2102.08981

[59] L. Chen, J. Li, X. Dong, P. Zhang, C. He, J. Wang, F. Zhao, and D. Lin, "Sharegpt4v: Improving large multi-modal models with better captions," 2023. [Online]. Available: https://arxiv.org/abs/2311.12793

[60] F. Ebert, Y. Yang, K. Schmeckpeper, B. Bucher, G. Georgakis, K. Daniilidis, C. Finn, and S. Levine, "Bridge data: Boosting generalization of robotic skills with cross-domain datasets," 2021. [Online]. Available: https://arxiv.org/abs/2109.13396

[61] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen *et al.*, "Lora: Low-rank adaptation of large language models." *ICLR*, vol. 1, no. 2, p. 3, 2022.

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

[63] Y. Jiang, A. Gupta, Z. Zhang, G. Wang, Y. Dou, Y. Chen, L. Fei-Fei, A. Anandkumar, Y. Zhu, and L. Fan, "Vima: General robot manipulation with multimodal prompts," *arXiv preprint arXiv:2210.03094*, vol. 2, no. 3, p. 6, 2022.

[64] T. Osa, J. Pajarinen, G. Neumann, J. A. Bagnell, P. Abbeel, and J. Peters, "An algorithmic perspective on imitation learning," *Foundations and Trends® in Robotics*, vol. 7, no. 1-2, pp. 1–179, 2018.

[65] J. Luo, Z. Hu, C. Xu, Y. L. Tan, J. Berg, A. Sharma, S. Schaal, C. Finn, A. Gupta, and S. Levine, "Serl: A software suite for sample-efficient robotic reinforcement learning," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 16 961–16 969.

[66] Y. Li, X. Ma, J. Xu, Y. Cui, Z. Cui, Z. Han, L. Huang, T. Kong, Y. Liu, H. Niu *et al.*, "Gr-rl: Going dexterous and precise for long-horizon robotic manipulation," *arXiv preprint arXiv:2512.01801*, 2025.

[67] P. Intelligence, A. Amin, R. Aniceto, A. Balakrishna, K. Black, K. Conley, G. Connors, J. Darpinian, K. Dhabalia, J. DiCarlo *et al.*, "$\pi^*_{0.6}$: a vla that learns from experience," *arXiv preprint arXiv:2511.14759*, 2025.

[68] P. J. Ball, L. Smith, I. Kostrikov, and S. Levine, "Efficient online reinforcement learning with offline data," in *International Conference on Machine Learning*. PMLR, 2023, pp. 1577–1594.

[69] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[70] Y. Lipman, R. T. Chen, H. Ben-Hamu, M. Nickel, and M. Le, "Flow matching for generative modeling," *arXiv preprint arXiv:2210.02747*, 2022.

[71] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke, and S. Levine, "Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation," 2018. [Online]. Available: https://arxiv.org/abs/1806.10293

[72] G. Dulac-Arnold, D. Mankowitz, and T. Hester, "Challenges of real-world reinforcement learning," *arXiv preprint arXiv:1904.12901*, 2019.

[73] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering." *ACM Trans. Graph.*, vol. 42, no. 4, pp. 139–1, 2023.

[74] Y. Chebotar, A. Handa, V. Makoviychuk, M. Macklin, J. Issac, N. Ratliff, and D. Fox, "Closing the sim-to-real loop: Adapting simulation randomization with real world experience," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8973–8979.

[75] M. Torne, A. Simeonov, Z. Li, A. Chan, T. Chen, A. Gupta, and P. Agrawal, "Reconciling reality through simulation: A real-to-sim-to-real approach for robust manipulation," 2024. [Online]. Available: https://arxiv.org/abs/2403.03949

[76] L. Yen-Chen, P. Florence, J. T. Barron, A. Rodriguez, P. Isola, and T.-Y. Lin, "Inerf: Inverting neural radiance fields for pose estimation," 2021. [Online]. Available: https://arxiv.org/abs/2012.05877

[77] B. Mehta, M. Diaz, F. Golemo, C. J. Pal, and L. Paull, "Active domain randomization," in *Conference on Robot Learning*. PMLR, 2020, pp. 1162–1176.

[78] S. Cheng, L. Ma, Z. Chen, A. Mandlekar, C. Garrett, and D. Xu, "Generalizable domain adaptation for sim-and-real policy co-training," *arXiv preprint arXiv:2509.18631*, 2025.

[79] A. Mandlekar, S. Nasiriany, B. Wen, I. Akinola, Y. Narang, L. Fan, Y. Zhu, and D. Fox, "Mimicgen: A data generation system for scalable robot learning using human demonstrations," *arXiv preprint arXiv:2310.17596*, 2023.

[80] T. Yu, D. Quillen, Z. He, R. Julian, A. Narayan, H. Shively, A. Bellathur, K. Hausman, C. Finn, and S. Levine, "Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning," 2021. [Online]. Available: https://arxiv.org/abs/1910.10897

[81] T. Dai, J. Wong, Y. Jiang, C. Wang, C. Gokmen, R. Zhang, J. Wu, and L. Fei-Fei, "Automated creation of digital cousins for robust policy learning," *arXiv preprint arXiv:2410.07408*, 2024.

[82] C. Yu, Y. Wang, Z. Guo, H. Lin, S. Xu, H. Zang, Q. Zhang, Y. Wu, C. Zhu, J. Hu *et al.*, "Rlinf: Flexible and efficient large-scale reinforcement learning via macro-to-micro flow transformation," *arXiv preprint arXiv:2509.15965*, 2025.