Sim-and-Human Co-training for Data-Efficient and Generalizable Robotic Manipulation

Kaipeng Fang[1], Weiqing Liang[1], Yuyang Li[1], Ji Zhang[2], Pengpeng Zeng[3], Lianli Gao[1], Heng Tao Shen[3], Jingkuan Song[3,4]

[1]University of Electronic Science and Technology of China
[2]Southwest Jiaotong University [3]Tongji University [4]Shanghai Innovation Institute

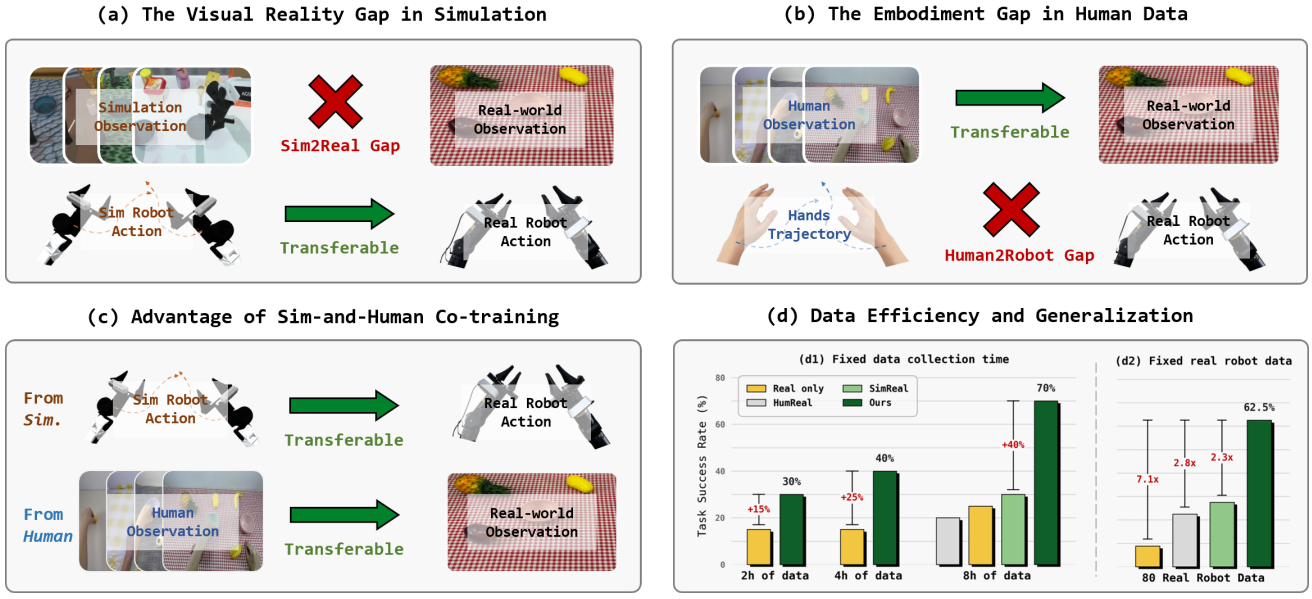

Fig. 1: **Sim-and-Human Co-training. (a)** Simulation data offers transferable actions but suffers from the Sim2Real gap; **(b)** human data provides realistic observations but is limited by the Human2Robot gap. **(c)** Our Sim-and-Human Co-training framework unifies transferable sim robot actions and human observations to mutually mitigate their respective gaps. **(d)** Consequently, our approach demonstrates exceptional data efficiency and generalization capabilities under OOD settings.

***Abstract*—Synthetic simulation data and real-world human data provide scalable alternatives to circumvent the prohibitive costs of robot data collection. However, these sources suffer from the *sim-to-real visual gap* and the *human-to-robot embodiment gap*, respectively, which limits the policy's generalization to real-world scenarios. In this work, we identify a natural yet underexplored complementarity between these sources: simulation offers the robot action that human data lacks, while human data provides the real-world observation that simulation struggles to render. Motivated by this insight, we present SimHum, a co-training framework to simultaneously extract kinematic prior from simulated robot actions and visual prior from real-world human observations. Based on the two complementary priors, we achieve data-efficient and generalizable robotic manipulation in real-world tasks. Empirically, SimHum outperforms the baseline by up to 40% under the same data collection budget, and achieves a 62.5% OOD success with only 80 real data, outperforming the real only baseline by $7.1\times$. Videos and additional information can be found at project website.**

I. Introduction

A long-standing challenge in robot learning is enabling policies to generalize across complex manipulation tasks in the unconstrained real-world. Recent advancements in robotic foundation models [1–9] have demonstrated that pre-training on large-scale robotic datasets is a promising paradigm for achieving such capability. However, unlike the readily available web-scale vision-language data, collecting diverse robot data in the physical world is labor-intensive and prohibitively expensive [10–13]. Such data constraints severely restrict the potential for scaling up robotic learning to achieve robust real-world generalization.

Recently, synthetic simulation data and human demonstrations have emerged as promising alternatives to mitigate this data scarcity. With the advent of automated data generation pipelines [14–20], acquiring robot data in simulation has become highly efficient, requiring minimal human effort. However, policies learned from simulation must overcome the sim-to-real gap, since the simulated visuals and physics do not perfectly align with the real world [21–26]. Alternatively, human demonstrations serve as another scalable data source, as they can be effortlessly collected via ubiquitous consumer devices [27–29]. Nevertheless, transferring skills

from human demonstrations to robots is challenging due to the inherent human-to-robot gap, induced by the kinematic mismatch between human hands and robotic grippers. While prior works address these gaps via explicit alignment [27–34] or unified co-training [27–29] [35–38, 26], they are often limited by complex design choices or the naive integration of heterogeneous data that neglects inherent domain distinctions. In contrast to seeking complex strategies to bridge the gap, we observe that these data sources are highly complementary. Simulation provides robot-valid actions absent in human data (Figure 1a), while human data offers real-world observations that are challenging to synthesize in simulation (Figure 1b). This complementary relationship leads us to ask:

*Can we achieve data-efficient real-world generalization by **simply co-training** on these two data sources?*

In this paper, we introduce a **Sim**-and-**Hum**an Co-training (**SimHum**) approach to answer the question. The core idea of our SimHum approach is to extracts transferable robot kinematic priors from simulated robot action and visual priors from human observation to enable generalizable real-world manipulation, as shown in Figure 1(c). To achieve this, we first establish a tailored data collection pipeline aimed at aligning robot actions in simulation and the observations in human demonstrations with the real robot's action and observation spaces, respectively. Then, we design a modular diffusion policy architecture to extracts transferable priors from collected simulation and human data through joint pre-training. Finally, we restructure the pre-trained policy by selectively retaining the modules containing transferable priors while discarding components incompatible with the real robot. Leveraging these valuable priors, we achieve robust and generalizable manipulation in the real world by fine-tuning on only a limited set of real robot data.

We empirically evaluate SimHum in real-world environments with four manipulation tasks detailed in Section IV-B. SimHum ensures exceptional real-world data efficiency and advanced zero-shot generalizability. Firstly, as shown in Figure 1(d1), we observe that by reallocating half of the robot data collection time to gathering simulation and human data, SimHum achieves up to a $\mathbf{40}\%$ performance improvement over the baseline. Secondly, SimHum exhibits generalization to objects and scenes not encountered in any dataset, with relative improvements of up to $\mathbf{7.1}\times$ compared to Real only baseline (Figure 1, d2).

Our main contributions can be summarized as follows:

1) We propose Sim-and-Human Co-training (SimHum) to pre-train a unified policy exclusively on simulation and human data, establishing a scalable paradigm for data-efficient and generalizable robot manipulation.
2) Through systematic analysis, we identify the complementary roles of simulation and human data and establish a balanced co-training recipe for efficiently scaling heterogeneous datasets.
3) Extensive real-world evaluations demonstrate that SimHum achieves superior data efficiency and zero-shot generalization, substantially outperforming robot-only and single-source baselines.

II. Related Works

A. Imitation Learning for Robot Manipulation

Imitation Learning (IL), particularly in the form of Behavior Cloning (BC), has emerged as a dominant paradigm for acquiring robust visuomotor policies directly from expert demonstrations [39–49]. By mapping sensory observations to control actions, IL circumvents the complexities of manual controller design and reward engineering required by reinforcement learning. Recent advancements have demonstrated the remarkable scalability of this paradigm, driven largely by the evolution of policy architectures [1–9] and data collection systems [50–61]. However, the prohibitive cost of acquiring large-scale real-world data remains a fundamental bottleneck that limits the generalization of imitation learning policies. To overcome this, we propose a co-training framework that harmonizes the visual and kinematic priors from scalable human and simulation data to achieve data-efficient generalization.

B. Learning from Scalable Heterogeneous Sources

Leveraging scalable data sources, particularly simulation [14–20, 62–67] and human demonstrations [68–79], has become a standard practice to mitigate the scarcity of real-robot data. To harness these heterogeneous sources, prior works generally follow two primary paradigms: explicit domain alignment and unified policy co-training.

Explicit Domain Alignment. To align heterogeneous data for robot execution, many works focus on explicitly bridging distribution shifts. Methods in this category typically achieve alignment either through specific training objectives, such as Optimal Transport [26, 30] or MixUp interpolation [31], or by preprocessing data via retargeting [27, 32] and visual editing [29, 33, 34]. However, these approaches often assume that disparate domains must be forced to match, relying on intricate manual tuning or complex auxiliary losses that fundamentally limit their scalability.

Co-training for Unified Representation. In contrast to explicit alignment, a growing body of literature employs unified architectures to directly co-train on heterogeneous data, aiming to learn an embodiment-agnostic representation [26–29] [35–38]. By circumventing complex alignment designs, these methods effectively scale up policy learning with massive mixed datasets. However, they often treat simulation and human data merely as generic data augmentation to be pooled together, failing to strategically leverage the unique strengths inherent to each domain. In contrast to these unified approaches, our framework, **SimHum**, relies on the intrinsic *complementarity* of data sources. Instead of blindly pooling disparate domains, we explicitly *disentangle* and *extract* the reliable priors where each source excels—robot kinematics from simulation and visual realism from human data—structurally assembling a generalizable policy without expensive alignment.

III. METHOD

In this section, we present **SimHum** (Sim-and-Human Co-training), a novel approach for simultaneously leveraging scalable simulation and human data to enhance robotic policy learning.

A. Problem Formulation

We formulate robotic manipulation as a conditional sequence generation problem. Let $\mathcal{D}_{real} = \{(o_t, s_t, \mathbf{a}_{t:t+H})\}$ be a real-world robot dataset, comprising visual observations $o_t \in \mathbb{R}^{H\times W\times 3}$, proprioceptive states $s_t \in \mathbb{R}^{d_s}$, and the ground-truth action sequence $\mathbf{a}_{t:t+H} \in \mathbb{R}^{H\times d_a}$. Our goal is to learn a policy π_θ that approximates the expert distribution within $\mathcal{D}_{real}$. Since generalizable policies are inherently data-hungry, relying solely on costly real-world data is impractical. Consequently, recent works have leveraged scalable simulation and human data, despite their distinct distributional shifts:

Simulation Data ($\mathcal{D}_{sim}$). Simulation provides a kinematically aligned robot configuration where the robot action closely matches the real world, i.e., $\mathcal{A}_{sim} = \mathcal{A}_{real}$. However, due to rendering artifacts and simplification, the visual observation distribution suffers from a significant *sim-to-real visual gap*:

$$P_{sim}(o_t|s_t^{env}) \neq P_{real}(o_t|s_t^{env}) \tag{1}$$

where s_t^{env} denotes the environment state. This misalignment renders visual features learned purely from simulation unreliable for real-world deployment.

Human Data ($\mathcal{D}_{hum}$). Conversely, human data provides photorealistic visual observations that closely match the real-world distribution, $P_{hum}(o|s_t^{env}) \approx P_{real}(o|s_t^{env})$. However, the difference in morphology introduces a fundamental *human-to-robot embodiment gap*:

$$\mathcal{A}_{hum} \neq \mathcal{A}_{real} \tag{2}$$

Consequently, human actions cannot be directly executed by the robot without explicit retargeting or adaptation.

Objective. Instead of explicitly aligning these distributions, our objective is to learn a unified policy capable of extracting complementary priors from both sources—specifically, kinematic control priors from $\mathcal{D}_{sim}$ and visual semantic priors from $\mathcal{D}_{hum}$—thereby generalizing to the target domain $\mathcal{D}_{real}$ without requiring extensive real-world data.

B. Data Collection Pipeline

To guarantee that our heterogeneous data sources are directly transferable to the real robot, we implement a protocol that aligns the kinematic space of $\mathcal{D}_{sim}$ and the visual space of $\mathcal{D}_{hum}$ with the target robot setup. Specifically, $\mathcal{D}_{sim}$ guarantees kinematic consistency by employing robot URDFs identical to the real robot, thereby ensuring that the learned action priors are transferable. Similarly, $\mathcal{D}_{hum}$ secures visual alignment by capturing images using the same camera model from an identical viewpoint as the robot, guaranteeing that the learned visual priors align with the real-world deployment. Furthermore, to capture task-relevant manipulation priors, we collect data for an identical set of tasks across all three sources. Please refer to Appendix VII-E for additional details on data collection and processing.

C. Modular Policy Architecture

Our model architecture is constructed upon the DiT policy [80] where image observations are encoded by the vision encoder and the proprioceptive state is projected via a shallow MLP. The observation tokens and the noise token serve as input to an encoder-decoder transformer, which generate the denoising output. To effectively learn from both simulation and human data and extract transferable priors, we incorporate several modular components into the architecture. The overall architecture of our model is illustrated in Figure 2(a).

Modular Action Encoder and Decoder. We utilize the human encoder to project human hand poses into latent tokens, and the corresponding decoder to translate the predicted latent actions back into the human pose space. Similarly, we employ a robot-specific encoder and decoder to process robot proprioceptive states and control actions from simulation data in the same manner. This design enables the backbone to learn embodiment-invariant manipulation semantics in a unified latent space, while isolating embodiment-specific kinematic details within the encoder and decoder.

Domain-specific Vision Adaptors. For both simulation and human visual observations, we first employ a shared vision encoder to extract the initial visual tokens. Subsequently, we process the simulation tokens via a specific simulation visual adaptor, and the human tokens via a real-world visual adaptor, respectively. Structurally, both adaptors are designed as two-layer MLPs. This design enables the backbone to process a consistent visual representation, while effectively isolating domain-specific information within the respective adaptors.

D. Two-Stage Training Paradigm

In the following section, we detail our two-stage training paradigm, designed to extract valuable priors from simulation and human data, followed by an effective adaptation strategy for real-world deployment.

Sim-and-Human Pre-training. In this stage, we aim to simultaneously extract the robot kinematic prior from $\mathcal{D}_{sim}$ and the real-world visual prior from $\mathcal{D}_{hum}$ into our modular policy π_θ. To achieve this, we conduct joint training on both datasets, optimizing the policy via standard noise prediction objective for diffusion models, defined as:

$$\mathcal{L}(\theta; \mathcal{D}) = \mathbb{E}_{(o,a)\sim\mathcal{D},\epsilon\sim\mathcal{N}(0,I),t}\left[\|\epsilon - \epsilon_\theta(z_t, t, o)\|^2\right] \tag{3}$$

where $\epsilon \sim \mathcal{N}(0, I)$ represents the sampled Gaussian noise, z_t denotes the noisy latent action at timestep t, and ϵ_θ is the noise predicted by the policy. Based on this formulation, the total optimization objective is a weighted combination of losses from two data sources:

$$\mathcal{L}_{total} = (1-\alpha)\cdot\mathcal{L}(\theta; \mathcal{D}_{sim}) + \alpha\cdot\mathcal{L}(\theta; \mathcal{D}_{hum}) \tag{4}$$

where $\alpha \in [0, 1]$ is the co-training ratio balancing the relative weight of simulation and human data. Following [37], we implement α by controlling the data distribution within each

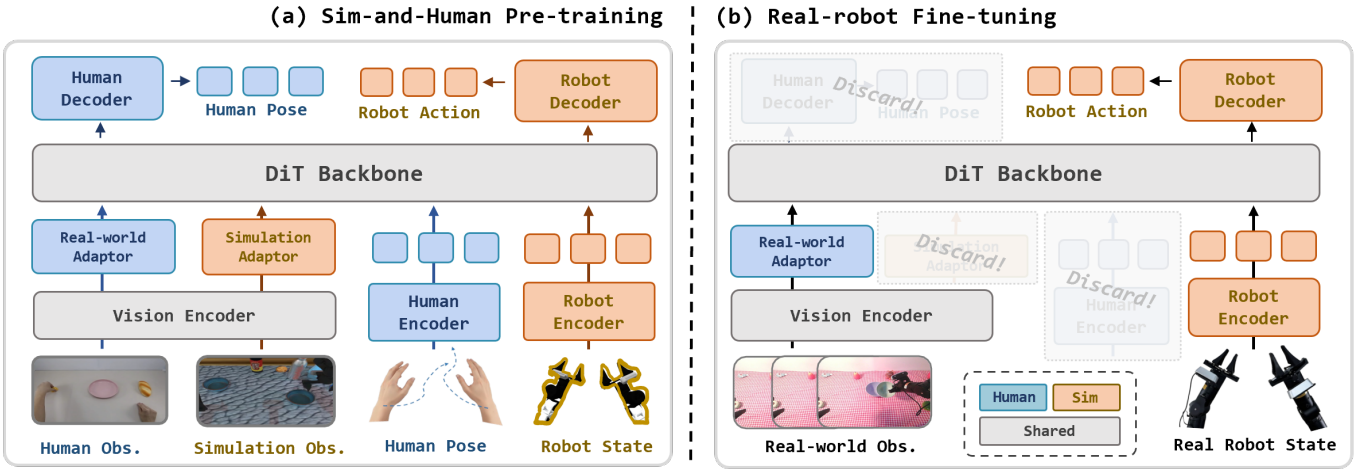

Fig. 2: **Model Overview.** Our approach consists of two main stages: (1) Sim-and-Human Pre-training that disentangles visual and action representations from simulation and human data to establish a generalized manipulation prior, and (2) Real-robot fine-tuning that couples the real-world adaptor with the action encoder-decoder to achieve data-efficient and generalizable real-world manipulation.

mini-batch. Specifically, for a batch size of B, we sample $\alpha \cdot B$ transitions from $\mathcal{D}_{hum}$ and $(1-\alpha) \cdot B$ from $\mathcal{D}_{sim}$, ensuring the effective gradient updates match the weighted objective. Experiments detailed in Section V-D suggest that the mixing ratio strongly affects final performance, with an equal weighting strategy ($\alpha = 0.5$) proving to be the most effective configuration in our pre-training setup.

Real-robot Fine-tuning. To adapt pre-trained model for the target robot in the real world, we reconstruct the policy architecture as depicted in Figure 2(b). Specifically, we compose the target policy using the Real-World Vision Adaptor from the human stream to preserve real-world visual priors, paired with the robot encoder/decoder to guarantee robot kinematic alignment. Finally, the model is fine-tuned on a small real-world dataset to adapt the components to the target robot. This selective recombination enables the policy to leverage the complementary strengths of both sources—transferable action priors from simulation and photorealistic visual priors from human data—thereby facilitating data-efficient generalization with minimal real-robot data.

IV. Experiment Setup

In this section, we detail the experimental setup designed to validate the proposed framework. We introduce the task suite, evaluation protocols, and the baseline methods used for comparison.

A. Data Composition Factors

To systematically analyze how specific dataset construction choices influence co-training efficacy, we decompose the datasets into discrete composition factors, following [37]. We assume that each dataset can be formulated as a joint distribution defined over these composition factors $\mathcal{F} = \{\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_k\}$. In this paper, we primarily focus on the following factors:

- **Target Object ($\mathcal{F}_{obj}$):** The specific object instance required to complete the task, characterized by its geometry and semantics.
- **Visual Distractors ($\mathcal{F}_{dist}$):** Task-irrelevant objects that introduce clutter and occlusion.
- **Lighting ($\mathcal{F}_{light}$):** Variations in lighting spectrum and intensity.
- **Background ($\mathcal{F}_{bg}$):** Workspace surface attributes, including texture and color.
- **Initial Pose ($\mathcal{F}_{init}$):** The distribution of the target object's initial 6-DoF pose.

Based on these factor definitions, we established the data collection (Section IV-C) and evaluation (Section IV-D) protocols. Furthermore, in Section V-B, we systematically investigate the impact of each factor's composition on model generalization.

B. Evaluation Tasks

We evaluate our approach on four diverse manipulation tasks, each designed to assess specific aspects of bimanual manipulation. Refer to Appendix VII-B for additional details about these tasks.

- **Stack Bowls Two**: Retrieves two bowls from different locations and stacks one atop the other, evaluating sequential object interaction and alignment.
- **Click Bell**: Accurately positions the gripper tip to actuate a small button on a service bell, evaluating fine-grained precision control.
- **Grab Roller**: Grasps a long roller stick simultaneously with both grippers to lift it horizontally, evaluating synchronized bimanual coordination.
- **Put Bread Cabinet**: Coordinates grasping a piece of bread with one arm while manipulating a drawer handle with the other for insertion, evaluating composite long-horizon planning.

C. Data Collection Protocol

Guided by the environment factors defined in Section IV-A, we establish the data collection protocol to systematically construct diverse training scenarios. The data collection protocol for each source is summarized below:

- **Simulation Data.** Utilizing the RoboTwin2.0 [19] framework, we generated 500 kinematically feasible trajectories per task. We applied the framework's default Domain Randomization techniques to enhance scenarios diversity.
- **Human Data.** We collected 500 human demonstrations for each task across 12 different scenarios. This diverse dataset allows the model to acquire robust photorealistic visual priors from the real world.
- **Real-world Data.** For fine-tuning, we strictly constrained the dataset to 80 episodes: 50 from a base environment and 30 from three complex scenarios (10 episodes each). All models are fine-tuned using this identical dataset to ensure a fair comparison.

The data collection scenes are constructed by varying the factor configuration $\{\mathcal{F}_{obj}, \mathcal{F}_{bg}, \mathcal{F}_{light}, \mathcal{F}_{dist}\}$, while strictly enforcing randomized object initialization $\mathcal{F}_{init}$ for each episode. For comprehensive details regarding the scenes configuration and initialization workspaces of each task, please refer to Appendix VII-E.

D. Evaluation Protocol

To quantitatively assess policy performance, we report the Success Rate (SR) to measure final task completion and Progress Rate (PR) to quantify the completion percentage of sub-goals, with detailed mathematical formulations outlined in Appendix VII-C. Specifically, we evaluate the model's performance under two distinct settings.

- **In-Distribution (ID) Setting.** This setting evaluates the policy across four specific scenarios encountered in the real-robot dataset, comprising a canonical Base Scenario (clean background, stable lighting) and three Complex Scenarios featuring domain variations such as diverse textures, moderate distractors, and dynamic lighting interference. We conduct 10 randomized trials per scenario, totaling 40 evaluation trials.
- **Out-of-Distribution (OOD) Setting.** To evaluate zero-shot generalization, this setting features two "extreme complex" scenarios synthesized by varying the data factors. Unlike the ID setting, these scenarios impose high-density clutter and irregular background textures using strictly held-out instances never seen in all datasets. We perform 10 trials per scenario with randomized initialization, totaling 20 trials.

For detailed environmental configurations and visual illustrations, please refer to Appendix VII-D.

E. Baselines

We evaluate **SimHum**—our full method pre-trained on simulation and human data—against three baseline configurations to verify the synergistic benefits of combining simulation and human priors:

- **Real only:** A policy initialized randomly and trained *from scratch* solely on the real-robot dataset. This serves as a lower bound to assess the necessity of pre-training.
- **SimReal:** A policy pre-trained on simulation data followed by fine-tuning on real-world data. This variant isolates the contribution of simulation priors and tests the direct sim-to-real transferability.
- **HumReal:** A policy pre-trained on human demonstrations. Due to the action space mismatch, the state encoder and action decoder are re-initialized before fine-tuning on real data. This variant isolates the contribution of human priors and evaluates the efficacy of learning from human demonstrations.

To ensure a fair comparison, we maintain strict consistency across all experimental settings. Specifically, all methods utilize the identical backbone architecture (see Appendix VII-I) and training hyperparameters (see Appendix VII-J). Furthermore, every variant is fine-tuned using the exact same set of real-robot data.

V. Results and Analysis

In this section, we empirically validate the proposed sim-and-human co-training framework. Our experiments are designed to answer the following four key research questions:

> *(1) How effectively does SimHum leverage simulation and human co-training to achieve robust real-world manipulation?*
> *(2) What specific priors does the policy leverage from simulation and human data, respectively?*
> *(3) Does SimHum demonstrate superior sample efficiency and scalability?*
> *(4) What key design choices yield the optimal performance for SimHum?*

A. Effectiveness of Sim-and-Human Co-training

SimHum enables robust policy generalization in novel scenes. To rigorously assess generalization capabilities, we evaluate policies under distinct In-Distribution (ID) and Out-of-Distribution (OOD) settings. As shown in Table I, We observe a progressive performance improvement across the evaluated methods. **First**, policies trained exclusively on real robot data (Real only) achieve basic performance in ID settings (40.0% SR and 59.4% PR) but degrade significantly in OOD scenarios (-31.2% SR and -27.7% PR). This performance collapse indicates that the policy relies on spurious visual correlations rather than learning invariant task representations. **Second**, single-source pre-training (SimReal/HumReal) yields restricted improvements, limiting SR gains to a range of $2.5\% \sim 7.5\%$ in ID and $13.7\% \sim 18.7\%$ in OOD settings. These methods are bottlenecked by their respective specific limitations: the visual gap in simulation and the embodiment

TABLE I: **Effectiveness of Sim-and-Human Co-training.** We compare the performance under In-Distribution and Out-of-Distribution settings as detailed in Section IV-D. We report the Success Rate (SR) and Progress Rate (PR) for all tasks. The values are presented as mean ± standard error.

Method	Stack Bowls Two		Click Bell		Put Bread Cabinet		Grab Roller		Average	
	SR ↑	PR ↑	SR ↑	PR ↑	SR ↑	PR ↑	SR ↑	PR ↑	SR ↑	PR ↑
In-Distribution Setting										
Real only	52.5 ± 8.0	74.2 ± 7.0	47.5 ± 8.0	65.0 ± 8.0	27.5 ± 7.0	48.3 ± 8.0	32.5 ± 7.0	50.0 ± 8.0	40.0 ± 3.9	59.4 ± 3.9
HumReal	57.5 ± 8.0	80.0 ± 6.0	55.0 ± 8.0	62.5 ± 8.0	30.0 ± 7.0	51.7 ± 8.0	27.5 ± 7.0	52.5 ± 8.0	42.5 ± 3.9	61.7 ± 3.8
SimReal	65.0 ± 8.0	86.7 ± 5.0	55.0 ± 8.0	73.8 ± 7.0	35.0 ± 8.0	55.0 ± 8.0	35.0 ± 8.0	61.3 ± 8.0	47.5 ± 3.9	69.2 ± 3.6
Ours	**77.5 ± 7.0**	**90.0 ± 5.0**	**70.0 ± 7.0**	**83.8 ± 6.0**	**65.0 ± 8.0**	**78.3 ± 7.0**	**57.5 ± 8.0**	**76.3 ± 7.0**	**67.5 ± 3.7**	**82.1 ± 3.2**
Out-of-Distribution Setting										
Real only	15.0 ± 8.0	48.3 ± 11.0	15.0 ± 8.0	35.0 ± 11.0	5.0 ± 5.0	23.3 ± 9.0	0.0 ± 0.0	20.0 ± 9.0	8.8 ± 3.2	31.7 ± 5.1
HumReal	30.0 ± 10.0	58.3 ± 11.0	25.0 ± 10.0	42.5 ± 11.0	20.0 ± 9.0	45.0 ± 11.0	15.0 ± 8.0	27.5 ± 10.0	22.5 ± 4.7	43.3 ± 5.4
SimReal	40.0 ± 11.0	60.0 ± 11.0	40.0 ± 11.0	52.5 ± 11.0	20.0 ± 9.0	41.7 ± 11.0	10.0 ± 7.0	40.0 ± 11.0	27.5 ± 5.0	48.5 ± 5.5
Ours	**75.0 ± 10.0**	**83.3 ± 8.0**	**60.0 ± 11.0**	**80.0 ± 9.0**	**55.0 ± 11.0**	**68.3 ± 10.0**	**60.0 ± 11.0**	**77.5 ± 9.0**	**62.5 ± 5.4**	**77.3 ± 4.5**

gap in human data. **Finally**, by simultaneously extracting transferable priors from both simulation and human data, SimHum achieves superior performance across all tasks. Most notably in OOD scenarios, it surpasses the single-source baseline by $+\mathbf{35.0}\%$ and the Real only policy by a factor of $\mathbf{7.1}\times$ on average SR , demonstrating SimHum's robust generalization capabilities in complex, novel scenes.

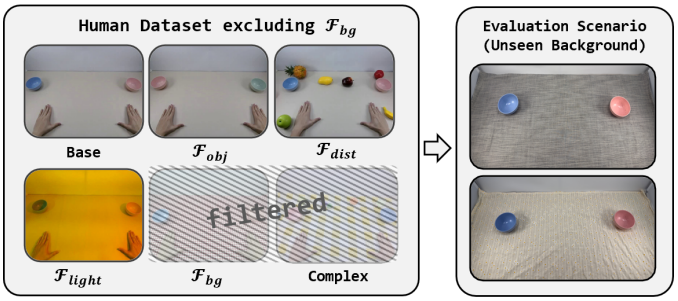

Fig. 3: **Experimental setup for evaluating the impact of the background factor ($\mathcal{F}_{bg}$) in human data.** We exclude $\mathcal{F}_{bg}$-related samples from the human data and design two corresponding OOD scenarios to quantify how this reduction in diversity affects the policy's ability to generalize to unseen backgrounds.

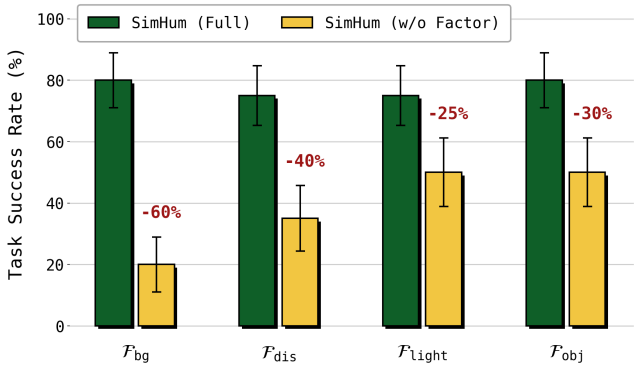

Fig. 4: **Human data enhances visual generalization across multiple aspects.** We compare SimHum on Stack Bowls Two trained with the full human dataset (Full) vs. subsets excluding specific factors (w/o Factor). Using the background factor ($\mathcal{F}_{bg}$) as a representative example, Figure 3 illustrates the workflow for filtering human data and designing the corresponding OOD evaluation scenarios.

B. Decoupling the Effects of Simulation and Human data

To investigate the distinct impact of priors extracted from simulation and human data on policy generalization, we conduct a fine-grained ablation study on the *Stack_Bowls_Two* task, yielding the following insights:

Human data is critical for visual generalization. To investigate the specific contribution of human data, we conduct an ablation study based on the data factors $\{\mathcal{F}_{obj}, \mathcal{F}_{bg}, \mathcal{F}_{light}, \mathcal{F}_{dist}\}$. Figure 3 illustrates the experimental setup for ablating the background factor ($\mathcal{F}_{bg}$): we exclude $\mathcal{F}_{bg}$-related diversity from human data and assess the trained policy in a corresponding OOD scenario with unseen backgrounds. Additional details are provided in Appendix VII-F. Based on the results shown in Figure 4, we observe that removing any single factor leads to a consistent decline in policy success rates within the target OOD scenarios. Notably, the exclusion of background diversity resulted in the most severe performance degradation ($-\mathbf{60}\%$ in SR). We attribute this to the fact that background variations inducing the most significant visual perturbations in robot observations. Overall, the diversity across different dimensions in human data equips the model with a robust visual prior, enabling generalization across complex, real-world scenarios.

Simulation data enhances policy robustness to novel object positions. In this experiment, we fine-tuned SimHum and HumReal on 80 real robot episodes where initial object positions were strictly confined to a fixed 3×3 grid. For evaluation, objects were positioned across an expanded 4×4 grid within an OOD scenario. We report the average Progress Rate (PR) per grid location, aggregated over 10 trials. Experimental results in Figure 5 indicate that while HumReal is capable in seen positions (achieving 60.7% PR on average), it struggles with spatial generalization, marked by a $-\mathbf{36.9}\%$

PR decay in unseen regions. In contrast, SimHum leverages diverse robotic action trajectories from simulation data, thereby achieving robustness in seen zones with a $+\mathbf{23}\%$ improvement and extending generalization capabilities to unseen areas by $+\mathbf{36.7}\%$ compared to HumReal. This demonstrates that the kinematic diversity inherent in simulation is critical for generating feasible, IK-solvable robotic actions even when the model encounters objects in novel positions.

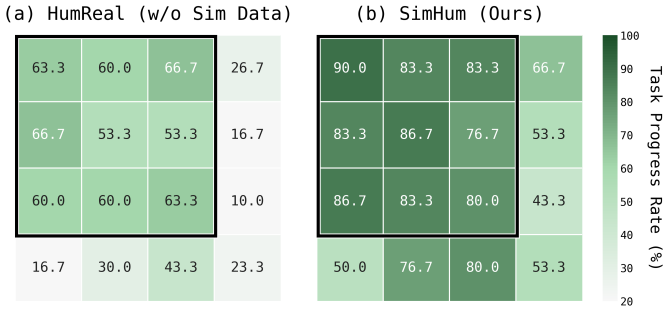

Fig. 5: **Simulation data enhances the generalization to unseen positions.** We evaluate position generalization by comparing HumReal and SimHum on a discretized 4×4 grid. For each position, we perform 10 evaluation trials and report the average Progress Rate. The black box outlines the region covered by real-robot training data, while the exterior represents uncovered positions.

C. Data Efficiency and Performance Scalability

SimHum achieves better data efficiency in limited data collection budgets. In this experiment, we investigate whether SimHum can effectively leverage simulation and human data to achieve better OOD performance under limited data collection budgets. We evaluated two primary strategies across total time budgets of 2, 4, and 8 hours: (1) Real only, where all time is spent on real data; and (2) SimHum, which splits the budget—50% for real data and 50% for the parallel collection of simulation and human data. For the 8-hour setting, we further evaluated (3) SimReal (4h real + 4h sim) and (4) HumReal (4h real + 4h human) to investigate the specific contribution of each data source. Detailed data configurations are listed in Appendix VII-H. As observed in Figure 6, the results highlight two key findings. **First,** SimHum consistently outperforms the Real only baseline across all time budgets, achieving improvements of up to +45%. This demonstrates that SimHum is significantly more data-efficient, outperforming Real only strategies within the same time budgets. **Second,** under the 8-hour setting, SimHum significantly surpasses both single-source baselines, exceeding SimReal by +40% and HumReal by +50%. This confirms that by effectively combining the complementary strengths of both data sources, SimHum achieves a synergistic effect that significantly enhances policy generalization in real-world.

Policy performance scales with real robot dataset size. To understand how real robot data scale impacts OOD performance, we vary the number of demonstrations from 8 to 160

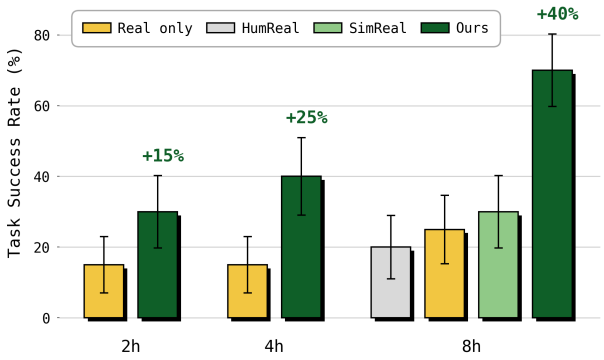

Fig. 6: **SimHum achieves better data efficiency in limited data collection budgets.** We evaluate policies under fixed data collection time limits. The Real only baseline dedicates the entire time budget to real robot data collection. In contrast, SimReal and HumReal allocate their time equally between real robot data and simulation or human data, respectively, whereas SimHum integrates all three sources—simulation, human, and real robot data. Please refer to Appendix VII-H for detailed data configurations.

and evaluate the resulting policies. As shown in Figure 7(a), SimHum consistently outperforms the Real only baseline at all scales. Remarkably, SimHum with only 8 real demonstrations achieves performance comparable to the Real only baseline trained with 160 demonstrations. This demonstrates that by leveraging pre-training on simulation and human data, SimHum facilitates significantly more efficient utilization of real robot data.

Policy performance scales with pre-train dataset size. In this experiment, we systematically investigate the impact of Simulation and Human dataset sizes on SimHum's generalization performance. To this end, we employ a controlled protocol: holding the size of one data source constant at 500 episodes while incrementally scaling its counterpart from 0 to 500. As illustrated in Figure 7(b), we observe that increasing the size of either the Human or Simulation dataset independently results in consistent performance improvements. This demonstrates that each data source offers valuable priors during pre-training, confirming that their combination is indispensable for real-world generalization. Furthermore, the performance trend remains upward even at 500 samples, suggesting that continued scaling of both data sources could yield further improvements.

Simulation and human data collection are faster and more scalable than robot teleoperation. We compare the data collection speed across different sources. As shown in Figure 8, collecting simulation or human data is approximately **5**$\times$ **faster** than real robot teleoperation on average. Furthermore, both simulation and human data circumvent the requirement for physical robots and complex data collection setups, thereby offering superior scalability.

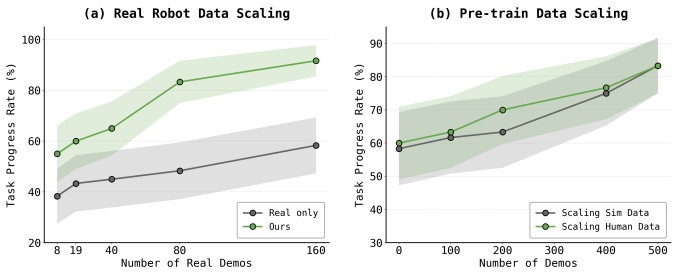

Fig. 7: **(a) Real robot data scaling.** We compare the Real only with SimHum when fine-tuned on varying scales of real robot data. **(b) Pre-train data scaling.** We pre-train SimHum with varying sizes of either the simulation or human data (ranging from 0 to 500), while holding the counterpart source constant at 500. All Experiments are conducted on the Stack Bowls Two task under OOD settings. The shaded regions denote the standard deviation.

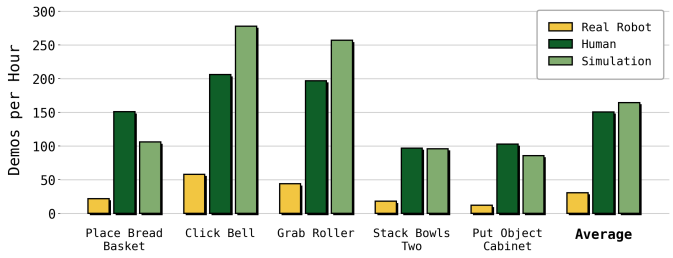

Fig. 8: **Human and simulation pipelines are 5× faster than robot teleoperation.** We compare the collection speeds of Real Robot, Human and Simulation data across five tasks. Note that simulation speed is evaluated using a single-threaded process on an NVIDIA RTX 4090 GPU.

D. Ablation Study on SimHum

In this section, we explore critical design choices regarding our model architecture and co-training strategies. All experiments are conducted on the Stack Bowls Two task.

Impact of SimHum Components. We quantify the contribution of each architectural design component in Table II. First, removing the real-world adaptor during fine-tuning results in a significant performance degradation (SR $-\mathbf{20}\%$, PR $-\mathbf{13}\%$). This demonstrates the critical necessity of the real-world vision adaptor for successfully transferring photorealistic visual priors to the real robot. Then, replacing the relative action with absolute action leads to a substantial decline (SR $-\mathbf{30}\%$, PR $\mathbf{25}\%$), validating its essential role in unifying the distinct absolute coordinate systems of heterogeneous data.

Impact of Co-training ratio. In this experiment, we investigate the impact of the co-training ratio α, which controls the proportion of human data in each batch, on the model's performance across both in-distribution (ID) and out-of-distribution (OOD) settings. As illustrated in Figure 9, the model achieves the best performance across both ID and OOD settings when $\alpha = 50\%$. This suggests that simulation data and human data are of equal importance during the pre-training phase. Furthermore, as α increases from 50% to 90%, we observe a simultaneous performance decline in both ID and OOD settings. We attribute this to the dominance of human data in the batch, which biases the model towards human-specific motion patterns that are difficult to transfer to the robot's control policy. Conversely, when α decreases from 50% to 10%, the degradation in OOD performance is notably more severe than in ID. This suggests that human data contributes critical visual priors that are essential for generalization to novel environments.

TABLE II: **Ablation Study of SHC Components.** We evaluate each design module on the Stack Bowls Two task under OOD setting, reporting Success Rate (SR) and Progress Rate (PR). The values are presented as mean ± standard error.

Method	**SR** ↑	**PR** ↑
SHC w/o real world adaptor	40.0 ± 15.5	68.3 ± 14.7
SHC w/o relative action	45.0 ± 15.7	58.3 ± 15.6
SHC (Ours)	$\mathbf{75.0 \pm 13.7}$	$\mathbf{83.3 \pm 11.8}$

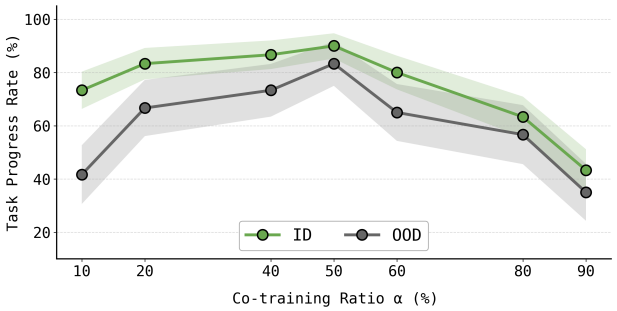

Fig. 9: **Effectiveness of the different co-training ratios.** The co-training ratio α controls the proportion of human data within a batch. We report the Progress Rate of SimHum on the Stack Bowls Two task under ID and OOD settings across varying α values, with shaded regions denoting standard deviation.

VI. Conclusion and Limitations

In this paper, we identified and systematically explored the underexplored complementarity between synthetic simulation data and real-world human demonstrations. Motivated by this discovery, we designed SimHum, a unified co-training framework for data-efficient and generalizable robotic manipulation. By explicitly integrating the robot kinematic priors from simulation with the photorealistic visual priors from human data, our approach effectively overcomes the limitations inherent in isolated data domains. Empirical results validate the efficacy of this synergy: SimHum outperforms the Real only baseline by 40% under a fixed 8-hour data collection budget and achieves a 62.5% success rate in OOD scenarios. Furthermore, we establish a balanced co-training recipe, offering actionable insights for scaling heterogeneous robot learning.

Despite these advancements, we identify four limitations guiding future work. First, current evaluations are restricted

to basic manipulation tasks; future research should extend to complex scenarios like dexterous manipulation and extremely long-horizon tasks. Second, while robustness is improved, the remaining OOD performance gap warrants further investigation to ensure reliability. Third, our current single-task policy lacks multi-tasking capabilities. Integrating our approach with Vision-Language-Action (VLA) models could enable open-world instruction following and cross-task transfer. Finally, moving beyond curated datasets to diverse in-the-wild data offers a promising path for exploring emergent behaviors in robotic foundation models.

REFERENCES

[1] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn, "Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware," in *Robotics: Science and Systems (RSS)*, 2023. 1, 2

[2] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song, "Diffusion Policy: Visuomotor Policy Learning via Action Diffusion," in *Robotics: Science and Systems (RSS)*, 2023.

[3] 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," in *Robotics: Science and Systems (RSS)*, 2023.

[4] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. D. Arenas, A. Dubey, C. Finn *et al.*, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," in *Conference on Robot Learning (CoRL)*, 2023.

[5] 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 *Robotics: Science and Systems (RSS)*, 2024.

[6] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. P. Foster, P. R. Sanketi, Q. Vuong, T. Kollar, B. Burchfiel, R. Tedrake, D. Sadigh, S. Levine, P. Liang, and C. Finn, "OpenVLA: An open-source vision-language-action model," in *Conference on Robot Learning (CoRL)*, 2024.

[7] S. Liu, L. Wu, B. Li, H. Tan, H. Chen, Z. Wang, K. Xu, H. Su, and J. Zhu, "RDT-1b: a diffusion foundation model for bimanual manipulation," 2025.

[8] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter *et al.*, "π_0: A Vision-Language-Action flow model for general robot control," *arXiv preprint arXiv:2410.24164*, 2024.

[9] K. Black, N. Brown, J. Darpinian, K. Dhabalia, D. Driess, A. Esmail, M. R. Equi, C. Finn, N. Fusai, M. Y. Galliker, D. Ghosh, L. Groom, K. Hausman, brian ichter, S. Jakubczak, T. Jones, L. Ke, D. LeBlanc, S. Levine, A. Li-Bell, M. Mothukuri, S. Nair, K. Pertsch, A. Z. Ren, L. X. Shi, L. Smith, J. T. Springenberg, K. Stachowicz, J. Tanner, Q. Vuong, H. Walke, A. Walling, H. Wang, L. Yu, and U. Zhilinsky, "$\pi_{0.5}$: a Vision-Language-Action model with Open-World generalization," in *Conference on Robot Learning (CoRL)*, 2025. 1, 2

[10] 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," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2024. 1

[11] A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, S. Dasari, S. Karamcheti, S. Nasiriany, M. K. Srirama, L. Y. Chen, K. Ellis, P. D. Fagan, J. Hejna, M. Itkina, M. Lepert, Y. J. Ma, P. T. Miller, J. Wu, S. Belkhale, S. Dass, H. Ha, A. Jain, A. Lee, Y. Lee, C. Lynch, V. Jain *et al.*, "Droid: A large-scale in-the-wild robot manipulation dataset," *arXiv preprint arXiv:2403.12945*, 2024.

[12] Q. Bu, J. Cai, L. Chen, X. Cui, Y. Ding, S. Feng, X. He, X. Huang *et al.*, "Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. in 2025 ieee," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025.

[13] S. Wu, X. Liu, S. Xie, P. Wang, X. Li, B. Yang, Z. Li, K. Zhu, H. Wu, Y. Liu *et al.*, "Robocoin: An open-sourced bimanual robotic data collection for integrated manipulation," *arXiv preprint arXiv:2511.17441*, 2025. 1

[14] A. Raistrick *et al.*, "Infinite photorealistic worlds using procedural generation," in *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*, 2023. 1, 2

[15] Y. Xu, W. Wan, J. Zhang, H. Liu, Z. Shan *et al.*, "Unidexgrasp: Universal robotic dexterous grasping via learning diverse proposal generation," in *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*, 2023.

[16] R. Wang, J. Zhang, J. Chen, Y. Xu, P. Li, T. Liu, and H. Wang, "Dexgraspnet: A large-scale robotic dexterous grasp dataset for general objects based on simulation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

[17] 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," in *Conference on Robot Learning (CoRL)*, 2023.

[18] Y. Mu, T. Chen, Z. Chen, S. Peng, Z. Lan, Z. Gao, Z. Liang, Q. Yu, Y. Zou, M. Xu *et al.*, "RoboTwin: Dual-arm robot benchmark with generative digital twins," in *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*, 2025, pp. 27 649–27 660.

[19] T. Chen, Z. Chen, B. Chen, Z. Cai, Y. Liu, Q. Liang, Z. Li, 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. 5, 15, 16

[20] Z. Jiang, Y. Xie, K. Lin, Z. Xu, W. Wan, A. Mandlekar, L. J. Fan, and Y. Zhu, "DexMimicGen: Automated data generation for bimanual dexterous manipulation via imitation learning," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2025, pp. 16 923–16 930. 1, 2

[21] A. Kumar, Z. Fu, D. Pathak, and J. Malik, "Rma: Rapid motor adaptation for legged robots," in *Robotics: Science and Systems (RSS)*, 2021. 1

[22] S. Ha, P. Xu, Z. Tan, S. Levine, and J. Tan, "Learning to walk in the real world with minimal human effort," in *Conference on Robot Learning (CoRL)*, 2020.

[23] Y. Shirai, K. Ota, D. K. Jha, and D. Romeres, "Sim-to-Real Contact-Rich Pivoting via Optimization-Guided RL with Vision and Touch," in *Conference on Neural Information Processing Systems (NeurIPS)*, 2025.

[24] W. Yu, D. Jain, A. Escontrela, A. Iscen, P. Xu, E. Coumans, S. Ha, J. Tan, and T. Zhang, "Visual-locomotion: Learning to walk on complex terrains with vision," in *Conference on Robot Learning (CoRL)*, 2022.

[25] Y. Qin, B. Huang, Z.-H. Yin, H. Su, and X. Wang, "Dexpoint: Generalizable point cloud reinforcement learning for sim-to-real dexterous manipulation," in *Conference on Robot Learning (CoRL)*, 2022.

[26] S. Cheng, L. Ma, Z. Chen, A. Mandlekar, C. R. Garrett, and D. Xu, "Generalizable domain adaptation for sim-and-real policy co-training," in *Conference on Neural Information Processing Systems (NeurIPS)*, 2025. 1, 2

[27] R.-Z. Qiu, S. Yang, X. Cheng, C. Chawla, J. Li, T. He, G. Yan, D. J. Yoon, R. Hoque, L. Paulsen *et al.*, "Humanoid policy ˜ human policy," *Conference on Robot Learning (CoRL)*, 2025. 1, 2

[28] T. Tao, M. K. Srirama, J. J. Liu, K. Shaw, and D. Pathak, "DexWild: Dexterous human interactions for in-the-wild robot policies," 2025.

[29] S. Kareer, D. Patel, R. Punamiya, P. Mathur, S. Cheng, C. Wang, J. Hoffman, and D. Xu, "EgoMimic: Scaling imitation learning via egocentric video," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2025, pp. 13 226–13 233. 1, 2

[30] R. Punamiya, D. Patel, P. Aphiwetsa, P. Kuppili, L. Y. Zhu, S. Kareer, J. Hoffman, and D. Xu, "Egobridge: Domain adaptation for generalizable imitation from egocentric human data," in *Human to Robot: Workshop on Sensorizing, Modeling, and Learning from Humans*, 2025. 2

[31] Y. Liu, W. C. Shin, Y. Han, Z. Chen, H. Ravichandar, and D. Xu, "Immimic: Cross-domain imitation from human videos via mapping and interpolation," *arXiv preprint arXiv:2509.10952*, 2025. 2

[32] R. Yang, Q. Yu, Y. Wu, R. Yan, B. Li, A.-C. Cheng, X. Zou, Y. Fang, X. Cheng, R.-Z. Qiu *et al.*, "EgoVLA: Learning vision-language-action models from egocentric human videos," *arXiv preprint arXiv:2507.12440*, 2025. 2

[33] M. Lepert, J. Fang, and J. Bohg, "Phantom: Training robots without robots using only human videos," in *Conference on Robot Learning (CoRL)*, 2025. 2

[34] M. Lepert, J. Fang, and J. Bohg, "Masquerade: Learning from in-the-wild human videos using data-editing," *arXiv preprint arXiv:2508.09976*, 2025. 2

[35] L. Wang, X. Chen, J. Zhao, and K. He, "Scaling proprioceptive-visual learning with heterogeneous pre-trained transformers," *Conference on Neural Information Processing Systems (NeurIPS)*, pp. 124 420–124 450, 2024. 2

[36] J. B. Nvidia, F. Castaneda, 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.

[37] A. Maddukuri, Z. Jiang, L. Y. Chen, S. Nasiriany, Y. Xie, Y. Fang, W. Huang, Z. Wang, Z. Xu, N. Chernyadev, S. Reed, K. Goldberg, A. Mandlekar, L. Fan, and Y. Zhu, "Sim-and-real co-training: A simple recipe for vision-based robotic manipulation," 2025. 3, 4

[38] 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," in *Robotics: Science and Systems (RSS)*, 2024. 2

[39] D. A. Pomerleau, "Alvinn: an autonomous land vehicle in a neural network," in *Conference on Neural Information Processing Systems (NeurIPS)*, 1988. 2

[40] T. Osa, J. Pajarinen, G. Neumann, J. A. Bagnell, P. Abbeel, and J. Peters, "An algorithmic perspective on imitation learning," *Found. Trends Robotics*, pp. 1–179, 2018.

[41] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robotics and Autonomous Systems (RAS)*, pp. 469–483, 2009.

[42] P. Florence, C. Lynch, A. Zeng, O. A. Ramirez, A. Wahid, L. Downs, A. Wong, J. Lee, I. Mordatch, and J. Tompson, "Implicit behavioral cloning," in *Conference on Robot Learning (CoRL)*, 2022.

[43] E. Jang, A. Irpan, M. Khansari, D. Kappler, F. Ebert, C. Lynch, S. Levine, and C. Finn, "Bc-z: Zero-shot task generalization with robotic imitation learning," in *Conference on Robot Learning (CoRL)*, 2022.

[44] S. Lee, Y. Wang, H. Etukuru, H. J. Kim, N. M. M. Shafiullah, and L. Pinto, "Behavior generation with latent actions," in *International Conference on Machine Learning (ICML)*, 2024, pp. 26 991–27 008.

[45] M. Shridhar, L. Manuelli, and D. Fox, "Perceiver-actor: A multi-task transformer for robotic manipulation," in *Conference on Robot Learning (CoRL)*, 2023.

[46] 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," in *International Conference on Machine Learning (ICML)*, 2023.

[47] A. Goyal, J. Xu, Y. Guo, V. Blukis, Y.-W. Chao, and D. Fox, "Rvt: Robotic view transformer for 3d object manipulation," in *Conference on Robot Learning (CoRL)*, 2023.

[48] S. Belkhale, Y. Cui, and D. Sadigh, "Hydra: Hybrid robot actions for imitation learning," in *Conference on Robot Learning (CoRL)*, 2023.

[49] S. Wu, X. Luo, J. Zhang, J. Xie, J. Song, H. T. Shen, and L. Gao, "Policy contrastive decoding for robotic foundation models," *arXiv preprint arXiv:2505.13255*, 2025. 2

[50] R. Zhang, T. Kim, M. Wen, N. Wu, and J. Wu, "Noir: Neural signal operated intelligent robots for everyday activities," in *Conference on Robot Learning (CoRL)*, 2023. 2

[51] Y. Qin, W. Yang, B. Huang, K. V. Wyk, H. Su, X. Wang, Y.-W. Chao, and D. Fox, "Anyteleop: A general vision-based dexterous robot arm-hand teleoperation system," in *Robotics: Science and Systems (RSS)*, 2023.

[52] Z. Fu, T. Z. Zhao, and C. Finn, "Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation," *arXiv preprint arXiv:2401.02117*, 2024.

[53] P. Wu, Y. Shentu, Z. Yan, E. Jang, and P. Abbeel, "Gello: A general, low-cost, and intuitive teleoperation framework for robot manipulation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[54] A. Iyer, Z. Peng, Y. Dai, A. Patel, S. Suresh, K. Shimada, and L. Pinto, "Open teach: A versatile teleoperation system for robotic manipulation," in *Conference on Robot Learning (CoRL)*, 2024.

[55] S. Yang, M. Liu, Y. Qin, R. Ding, J. Li, X. Cheng, R. Yang, S. Yi, and X. Wang, "Ace: A cross-platform visual-exoskeletons system for low-cost dexterous teleoperation," in *Conference on Robot Learning (CoRL)*, 2024.

[56] M. Ahn, D. Dwibedi, C. Finn, M. G. Arenas, K. Gopalakrishnan, K. Hausman, B. Ichter, A. Irpan, N. Joshi, R. Julian *et al.*, "Autort: Embodied foundation models for large scale orchestration of robotic agents," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[57] K. Liu, C. Guan, Z. Jia, Z. Wu, X. Liu, T. Wang, S. Liang, P. Chen, P. Zhang, H. Song *et al.*, "Fastumi: A scalable and hardware-independent universal manipulation interface with dataset," 2025.

[58] X. Cheng, J. Li, S. Yang, G. Yang, and X. Wang, "Open-television: Teleoperation with immersive active visual feedback," in *Conference on Robot Learning (CoRL)*, 2024.

[59] C. Wang, H. Shi, W. Wang, R. Zhang, L. Fei-Fei, and K. Liu, "DexCap: Scalable and portable mocap data collection system for dexterous manipulation," in *Robotics: Science and Systems (RSS)*, 2024.

[60] R. Ding, Y. Qin, J. Zhu, C. Jia, S. Yang, R. Yang, X. Qi, and X. Wang, "Bunny-visionpro: Real-time bimanual dexterous teleoperation for imitation learning," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025.

[61] Y. Ze, Z. Chen, J. P. Araujo, Z. ang Cao, X. B. Peng, J. Wu, and K. Liu, "TWIST: Teleoperated whole-body imitation system," in *Conference on Robot Learning (CoRL)*, 2025. 2

[62] F. Xiang, Y. Qin, K. Mo, Y. Xia, H. Zhu, F. Liu, M. Liu, H. Jiang, Y. Yuan, H. Wang *et al.*, "Sapien: A simulated part-based interactive environment," in *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*, 2020, pp. 11 094–11 104. 2

[63] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, and G. State, "Isaac Gym: High performance gpu based physics simulation for robot learning," 2021.

[64] C. D. Freeman, E. Frey, A. Raichuk, S. Girgin, I. Mordatch, and O. Bachem, "Brax: A differentiable physics engine for large scale rigid body simulation," *arXiv preprint arXiv:2106.13281*, 2021.

[65] J. Gu, F. Xiang, X. Li, Z. Ling, X. Liu, T. Sang, N. Seymour, X. Wei, and H. Su, "Maniskill2: A unified benchmark for generalizable manipulation skills," in *International Conference on Learning Representations (ICLR)*, 2023.

[66] S. Tao, F. Xiang, A. Shukla, Y. Qin, X. Hinrichsen, X. Yuan, C. Bao, X. Lin, Y. Liu, T.-K. Chan, Y. Gao, X. Li, T. Mu, N. Xiao, A. Gurha, V. N, Y. W. Choi, Y.-R. Chen, Z. Huang, R. Calandra, R. Chen, S. Luo, and H. Su, "ManiSkill3: GPU parallelized robot simulation and rendering for generalizable embodied AI," 2025.

[67] K. Zakka, B. Tabanpour, Q. Liao, M. Haiderbhai, S. Holt, J. Y. Luo, A. Allshire, E. Frey, K. Sreenath, L. A. Kahrs *et al.*, "MuJoCo Playground," *arXiv preprint arXiv:2502.08844*, 2025. 2

[68] O. Taheri, N. Ghorbani, M. J. Black, and D. Tzionas, "Grab: A dataset of whole-body human grasping of objects," in *European Conference on Computer Vision (ECCV)*, 2020. 2

[69] X. Chen, J. Hu, C. Jin, L. Li, and L. Wang, "Understanding domain randomization for sim-to-real transfer," 2022.

[70] F. Sener, D. Chatterjee, D. Shelepov, K. He, D. Chopra, B. Bhanu, and A. Gupta, "Assembly101: A large-scale multi-view video dataset for understanding procedural activities," in *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*, 2022.

[71] K. Grauman, A. Westbury, E. Byrne, Z. Chavis, A. Furnari, R. Giraldo, M. Hamburger, H. Hao, J. Pan, R. Piramuthu *et al.*, "Ego4D: Around the world in 3,000 hours of egocentric video," in *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*, 2022.

[72] Y. Liu, Y. Liu, C. Jiang, K. Lyu, W. Wan, H. Shen, B. Liang, Z. Fu, H. Wang, and L. Yi, "Hoi4d: A 4d egocentric dataset for category-level human-object interaction," in *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*, 2022.

[73] R. Khirodkar, A. Bansal, L. Ma, R. Newcombe, M. Vo, and K. Kitani, "Egohumans: An egocentric 3d multi-human benchmark," in *IEEE International Conference on Computer Vision (ICCV)*, 2023.

[74] O. Fan, Zicong andTaheri, D. Tzionas, M. Kocabas, S. Khamis, M. J. Black, O. Hilliges, and S. Tang, "Arctic: A dataset for dexterous bimanual hand-object manipulation," in *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*, 2023.

[75] P. Banerjee, S. Shkodrani, P. Moulon, S. Hampali, S. Han, F. Zhang, L. Zhang, J. Fountain, E. Miller, S. Basol *et al.*, "Hot3d: Hand and object tracking in 3d from egocentric multi-view," in *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*, 2025.

[76] D. Li, H. Zhang, Y. Ren, S. Zhao, J. Wang, and J. Li, "Openhumanvid: A large-scale high-quality dataset for enhancing human-centric video generation," in *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*, 2025.

[77] H. Chen, B. Sun, A. Zhang, M. Pollefeys, and S. Leutenegger, "Vidbot: Learning generalizable 3d actions from in-the-wild 2d human videos for zero-shot robotic manipulation," in *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*, 2025.

[78] K. Fan, J. Zhang, D. Wei, H. Zhang, X. Zhang, W. Li, M. Ren, Z. Zhang, X. Ye, J. Zhang *et al.*, "Go to zero: Towards zero-shot motion generation with million-scale data," in *IEEE International Conference on Computer Vision (ICCV)*, 2025.

[79] R. Hoque, P. Huang, D. J. Yoon, M. Sivapurapu, and J. Zhang, "Egodex: Learning dexterous manipulation from large-scale egocentric video," *arXiv preprint arXiv:2505.11709*, 2025. 2

[80] S. Dasari, O. Mees, S. Zhao, M. K. Srirama, and S. Levine, "The ingredients for robotic diffusion transformers," *arXiv preprint arXiv:2410.10088*, 2024. 3, 17

[81] C. Chi, Z. Xu, C. Pan, E. Cousineau, B. Burchfiel, S. Feng, R. Tedrake, and S. Song, "Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots," in *Robotics: Science and Systems (RSS)*, 2024. 16

VII. Appendix

A. Overview

The Appendix is organized as follows:

- **Task Definitions and Criteria** (Appendix VII-B): Details the definitions of the four manipulation tasks and their milestone-based scoring protocols.
- **Evaluation Metrics** (Appendix VII-C): Describes the quantitative metrics (Success Rate and Task Progress Rate) used for policy assessment.
- **Evaluation Settings** (Appendix VII-D): Specifies the In-Distribution (ID) and Out-of-Distribution (OOD) settings for evaluation.
- **Data Collection and Processing** (Appendix VII-E): Details the hardware systems, environmental protocols, and the data processing pipeline employed to simulation data, human data and real robot data.
- **Experimental Setup for Human Data Factor Ablation** (Appendix VII-F): Details the experimental setup of the human data factor ablation study presented in Section V-B.
- **Experimental Setup for Position Generalization Ablation** (Appendix VII-G): Outlines the experimental setup of the position generalization ablation presented in Section V-B.
- **Data Configurations for Time Budget Scaling Experiment** (Appendix VII-H): Provides the specific data volume and composition details for the time budget scaling experiments presented in Section V-C.
- **Implementation Details** (Appendix VII-I): Provides specific architectural details of the Source-dependent Modules, visual encoders, and the diffusion backbone.
- **Training Hyperparameters** (Appendix VII-J): Lists the exact optimization settings and training schedules for pre-training and fine-tuning.
- **Author Contributions** (Appendix VII-K): Lists the individual contributions of each author to the project.

B. Task Definitions and Scoring Criteria

We evaluate our proposed **SimHum** framework on four distinct manipulation tasks: Stack Bowls Two, Put Bread Cabinet, Click Bell, and Grab Roller. As visualized in Figure 10, these tasks are carefully selected to cover a wide range of motor skills, including bimanual coordination, long-horizon planning, precision actuation, and dynamic stability. To rigorously quantify the policy's performance beyond simple binary success, we designed a milestone-based scoring protocol for each task. This discretized scoring system allows us to distinguish between policies that fail completely and those that achieve partial success (e.g., successful grasping but failed manipulation), providing deeper insights into the specific capabilities learned by the model. Detailed task descriptions and scoring criteria are provided below:

Stack Bowls Two (Max Score: 3). This task requires precise dual-arm coordination to stack two bowls initially placed apart. The scoring reflects the cumulative success of independent arm actions and their coordination.

- **0:** Failure to grasp any bowl or meaningful interaction.
- **1:** Successfully grasping the left bowl with the left arm.
- **2:** Successfully grasping both bowls (one in each hand).
- **3:** Precisely stacking one bowl into the other without dropping either.

Put Bread Cabinet (Max Score: 3). This is a long-horizon task involving articulated object manipulation and multi-stage planning.

- **0:** Failure to grasp the bread object from the table.
- **1:** Successfully grasping the bread object.
- **2:** Successfully pull the cabinet door open.
- **3:** Placing the bread inside the cabinet and releasing it.

Click Bell (Max Score: 2). This task assesses the agent's fine motor control and precision in actuating a small target.

- **0:** Failure to reach the target area.
- **1:** Moving the end-effector to directly above the bell.
- **2:** Successfully applying downward force to depress the button and produce a ringing sound.

Grab Roller (Max Score: 2). This task evaluates bimanual synchronization and grasp stability on cylindrical objects.

- **0:** Failure to establish a stable grasp on the roller.
- **1:** Successfully establishing a grasp on the roller stick with both grippers.
- **2:** Lifting the roller to the target height with both arms while maintaining grasp without the object falling.

C. Evaluation Metrics

To measure both the final task completion and the quality of partial execution, we employ the following metrics. The specific milestone definitions and maximum scores (S_{max}) for each task are detailed in Appendix VII-B.

1) Success Rate (SR): SR measures the percentage of trials where the policy achieves the maximum possible score (i.e., complete task success). It is calculated as:

$$SR = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}(S_i = S_{max}) \times 100\% \tag{5}$$

where N is the total number of evaluation trials, S_i represents the score achieved in the i-th trial, S_{max} is the maximum score defined for the task, and $\mathbb{I}(\cdot)$ is the indicator function.

2) Task Progress Rate (PR): PR provides a fine-grained evaluation of the policy's ability to progress through multi-stage tasks, distinguishing between complete failures and partial successes. It is defined as the ratio of the total accumulated score across all trials to the maximum possible total score:

$$PR = \frac{\sum_{i=1}^{N} S_i}{N \times S_{max}} \times 100\% \tag{6}$$

A higher PR indicates that the policy consistently completes more stages of the task, even if it fails to achieve final success.

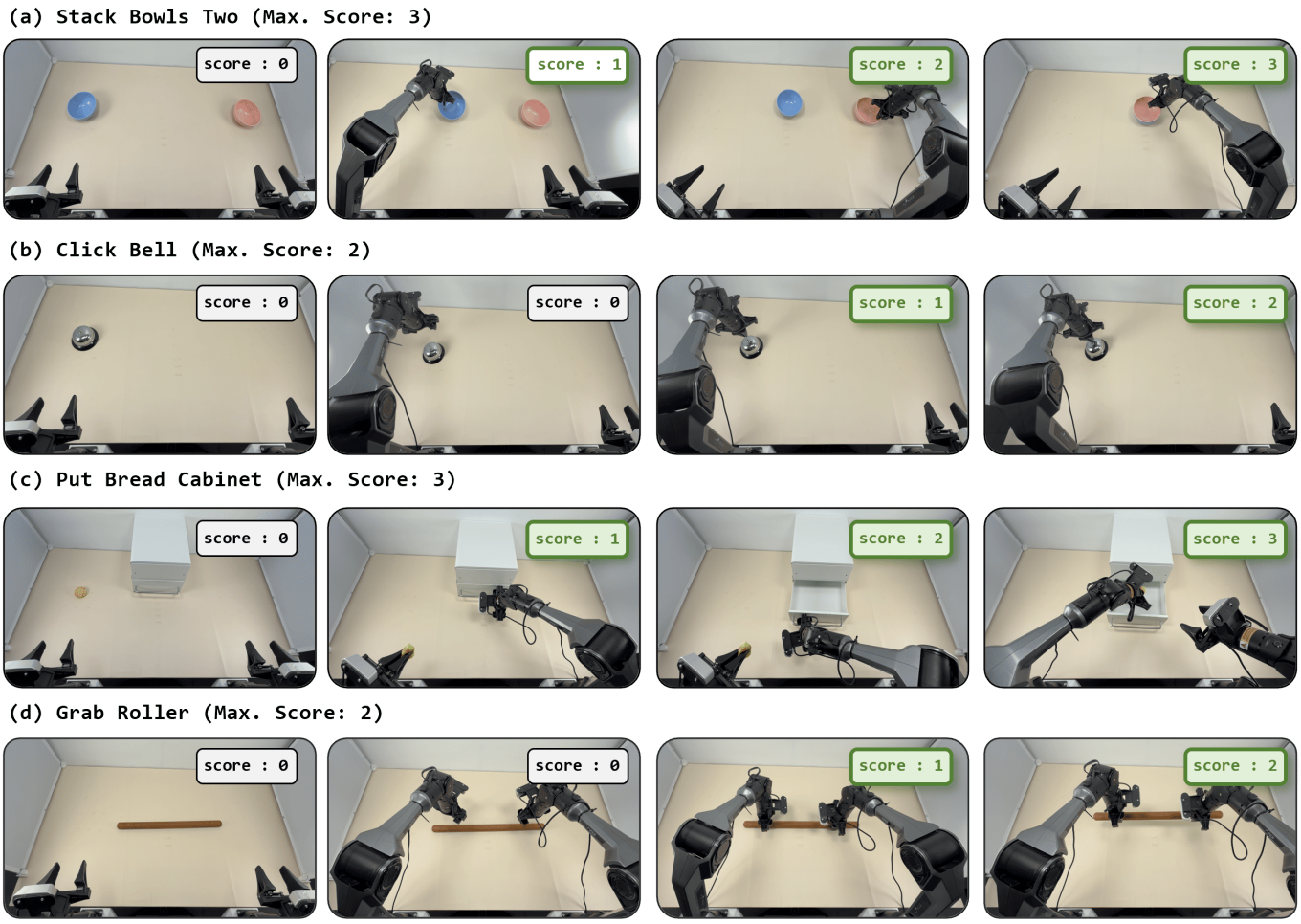

Fig. 10: **Task Definitions and Scoring.** Visual breakdown of the four manipulation tasks (Stack Bowls, Click Bell, Put Bread, Grab Roller) and their respective success/progress criteria.

D. Evaluation Settings

We classify the evaluation scenarios into In-Distribution (ID) and Out-of-Distribution (OOD) settings based on the environmental factors defined in Section IV-A. Visual examples of these settings are provided in Figure 11. For each evaluation trial, we randomly place the objects within the spatial region specified in Figure 15, which is consistent with the area used for data collection.

1) In-Distribution (ID): This setting evaluates the policy's performance in environments seen during the real-robot fine-tuning phase. As illustrated in Figure 11(a), it consists of two categories of scenarios:

- **Base Scenario (1 scenario):** A standard environment characterized by a clean white background and stable, neutral lighting conditions, serving as the baseline for robot operation.
- **Complex Scenarios (3 scenarios):** These scenarios introduce domain variations encountered during training. They feature backgrounds with diverse textures, a moderate number of distractor objects, and dynamic interference such as flashing blue lights.

Regarding the objects, the target items manipulated in the ID setting correspond strictly to the object instances seen during training, as depicted in the left panel of Figure 11(c). Crucially, all four ID scenarios appeared in the robot fine-tuning dataset. For each task, we conduct 10 evaluation trials per scenario, resulting in a total of 40 trials. We report the aggregated average SR and PR across these 40 trials.

2) Out-of-Distribution (OOD): To test zero-shot generalization, we construct two novel scenarios labeled as "Extreme Complex" (Figure 11(b)). In contrast to the ID settings, these scenarios are constructed entirely from strictly **held-out** environmental factors that never appeared in either the Human or Real-robot datasets. Specifically, we introduce novel background textures (featuring irregular wrinkles to increase complexity), unseen distractor objects arranged in higher density to create heavy clutter, and held-out task object instances (as highlighted in Figure 11(c)). We perform 10 evaluation trials for each of the two OOD scenarios, totaling 20 trials, and report the average SR and PR.

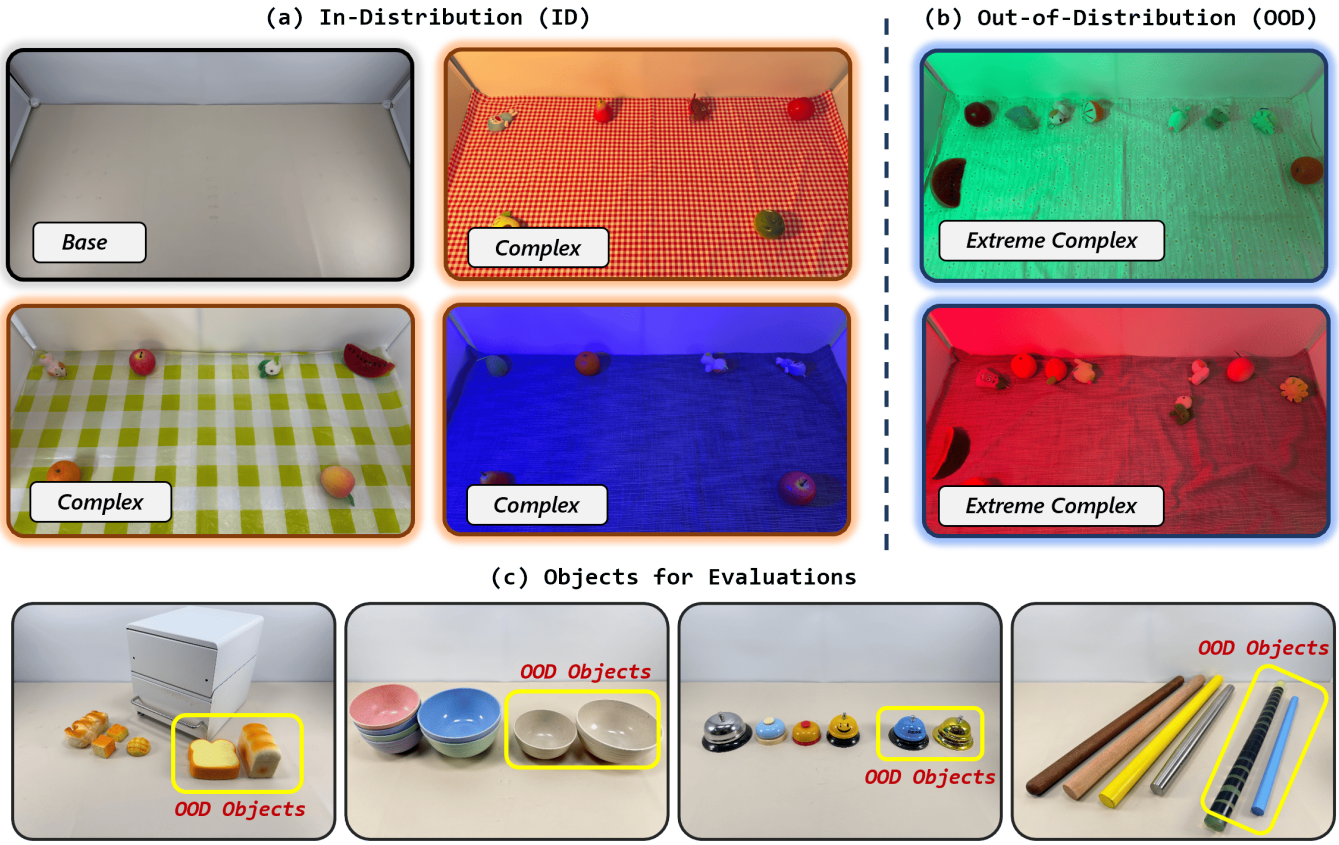

Fig. 11: **Evaluation Settings.** (a) **In-Distribution (ID)** scenarios comprising a standard *Base* environment and *Complex* environments seen during training. (b) **Out-of-Distribution (OOD)** scenarios labeled as *Extreme Complex*, constructed using strictly held-out environmental factors (e.g., unseen lighting, novel textures). (c) **Evaluation Objects** across tasks, where yellow bounding boxes indicate specific held-out instances (OOD Objects) introduced to test zero-shot generalization.

E. Data Collection and Processing Details

In this section, we provide a comprehensive overview of the data collection pipeline, covering the hardware systems, environmental protocols, and the data processing strategies employed to unify heterogeneous data sources.

Data Collection Systems. We utilize three distinct pipelines to acquire manipulation data, implementing a protocol that aligns the kinematic space of $\mathcal{D}_{sim}$ and the visual space of $\mathcal{D}_{hum}$ with the target robot to guarantee direct transferability, as illustrated in Figure 12:

- **Simulation Data Generation:** We leverage the RoboTwin 2.0 [19] to generate large-scale robot data. To ensure the applicability of this synthetic data, we employ robot assets that share identical kinematics with the physical hardware. We perform manipulation tasks strictly aligned with real-world specifications, yielding kinematically feasible trajectories that serve as robust robot action priors.
- **Real-Robot Data Collection:** For physical robot demonstrations, we utilize the COBOT Magic platform. Data is collected using a **leader-follower teleoperation** setup, where the operator controls the leader arms to drive the follower arms.
- **Human Data Collection (Right Panel):** To capture diverse human demonstrations, we designed a cost-effective ($<$ \$500) system aligned with the robot's observation space. We utilize an RGB camera identical to the robot's sensor, mounted on a stationary tripod to mimic the robot's static base perspective. A VR interface (Meta Quest 3) is used to capture high-precision human hand poses, while a recording pedal allows for hands-free control of the start/stop phases.

Environmental Protocols and Scenarios. To evaluate robustness, we strictly define the environmental factors for each domain. The object instances ($\mathcal{F}_{obj}$) are visualized in Figure 14, and valid spatial initialization ranges ($\mathcal{F}_{init}$) are shown in Figure 15. We construct distinct scenarios by combinatorially varying visual distractors ($\mathcal{F}_{dist}$), illumination ($\mathcal{F}_{light}$), and background ($\mathcal{F}_{bg}$):

- **Simulation Scenarios:** We employ the built-in Domain Randomization engine of RoboTwin to randomize textures, lighting, and distractors.(Figure 13(a)).
- **Human Scenarios:** To capture diverse real-world visual distributions, we curated 12 distinct scenarios by

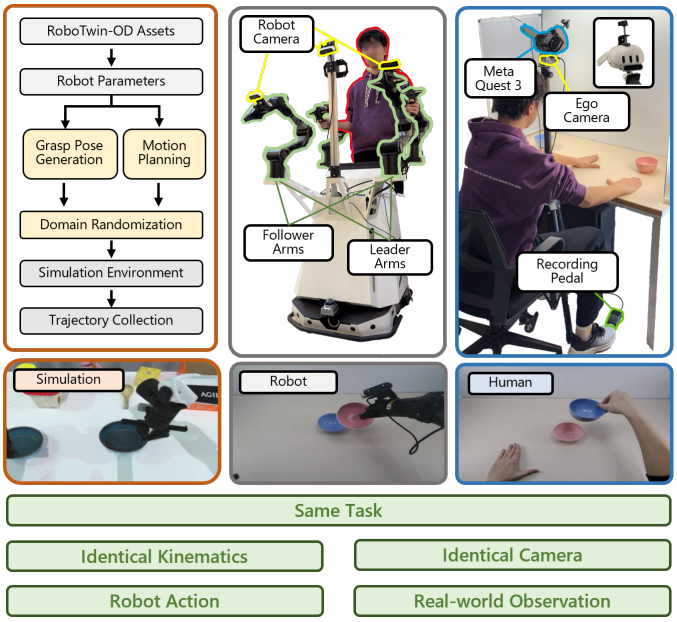

Fig. 12: **Overview of Data Collection Systems.** The figure illustrates the three data acquisition pipelines: (Left) The automated pipeline for Simulation data generation based on RoboTwin2.0 [19]; (Middle) The Real-robot teleoperation setup (COBOT Magic) employing a leader-follower teleoperation system; (Right) The Human data collection setup equipped with a VR interface and a stationary camera. The colored borders (Orange, Grey, Blue) correspond to the specific data source domains referenced in the text.

permuting environmental factors (Figure 13(b)). These setups encompass combinations of 4 tablecloth textures, 6 lighting conditions, and 16 distinct visual distractors to introduce realistic occlusion and clutter.

- **Real-Robot Scenarios:** To emulate a realistic data-scarce fine-tuning regime given the prohibitive cost of physical collection, we limited the design to 4 representative scenarios (1 base + 3 complex). As visualized in Figure 13(c), these utilize combinations of 3 tablecloths, 3 lighting conditions, and 12 visual distractors.

Action Space and Data Processing. To ensure the collected data is suitable for effective policy learning, we apply a standardized processing pipeline. First, regarding action representation, we formulate all actions as relative trajectories [81]. For robot and simulation data, we record a 16-dimensional vector ($a_t^R \in \mathbb{R}^{16}$) comprising the 14-dimensional end-effector pose (position and quaternion) and binary gripper states. Conversely, for human data, we utilize the wrist pose as the end-effector proxy and additionally record the 3D coordinates of fingertips relative to the wrist, yielding a 44-dimensional vector ($a_t^H \in \mathbb{R}^{44}$). Second, for temporal alignment, we downsample the human data (originally 30Hz) to 10Hz to match the robot's control frequency. Finally, we prune static segments from the beginning and end of each trajectory to eliminate idle periods caused by operator preparation, ensuring that only task-relevant interaction data is retained.

F. Experimental Setup for Human Data Factor Ablation

In this section, we detail the experimental protocol used to quantify the specific contributions of human data to visual generalization, as discussed in Section V-B. To rigorously isolate the impact of each data factor ($\mathcal{F}_{bg}$, $\mathcal{F}_{dist}$, $\mathcal{F}_{light}$, $\mathcal{F}_{obj}$), we designed a "Leave-One-Factor-Out" ablation strategy. The core logic of this experiment is illustrated in Figure 16. We systematically construct training subsets by filtering the full human dataset to exclude diversity associated with a specific target factor, thereby forcing the model to rely on a limited distribution for that dimension.

For a target factor (e.g., Background $\mathcal{F}_{bg}$), we first remove all human demonstrations that exhibit variations in that factor (e.g., excluding all trajectories collected on diverse tablecloths). The resulting training subset contains only the "base" condition for that specific factor (e.g., the default table surface) while retaining diversity in all other factors. Subsequently, we evaluate the policy trained on this restricted subset in unseen OOD scenarios that specifically feature the omitted factor. For instance, in the $\mathcal{F}_{bg}$ ablation, the model is evaluated on surfaces with novel textures (tablecloths) that were absent from its training distribution. This protocol ensures that any performance degradation compared to the full-data baseline can be directly attributed to the lack of visual priors for that specific environmental factor.

G. Experimental Setup for Position Generalization Analysis

In this section, we outline the experimental protocol used to evaluate the impact of simulation data on spatial robustness ($\mathcal{F}_{init}$). We first collected 80 real-robot demonstrations for the Stack Bowls Two task within a pre-defined 3×3 spatial grid, utilizing the visual setups consistent with Figure 13(c). For evaluation, we introduce an OOD scene and expand the workspace to a larger 4×4 grid. Since the Stack Bowls Two task necessitates simultaneous manipulation in both left and right workspaces, we establish two symmetric 4×4 initialization grids to govern the object placement. As illustrated in Figure 17, green zones denote the training coverage, while orange zones represent unseen positions. Evaluation is conducted across the full 4×4 grid, effectively probing the policy's performance on unseen outer coordinates (Orange). We conduct 10 independent evaluation trials for each of the 16 spatial positions and report the average Task Progress Rate.

H. Data Configurations for Time Budget Scaling

In this section, we provide the detailed data configurations for the Time Budget Scaling experiment discussed in Section V-C. Table III presents the exact volume of trajectories collected by each strategy (Real only, SimReal, HumReal, and SimHum) under total time budgets of 2, 4, and 8 hours. Notably, for our method, since simulation data generation is automated and does not require human intervention, it allows for parallel collection alongside human data acquisition. This parallel acquisition strategy enables **SimHum** to maximize data throughput and diversity within a fixed time budget, providing a highly data-efficient solution for policy learning.

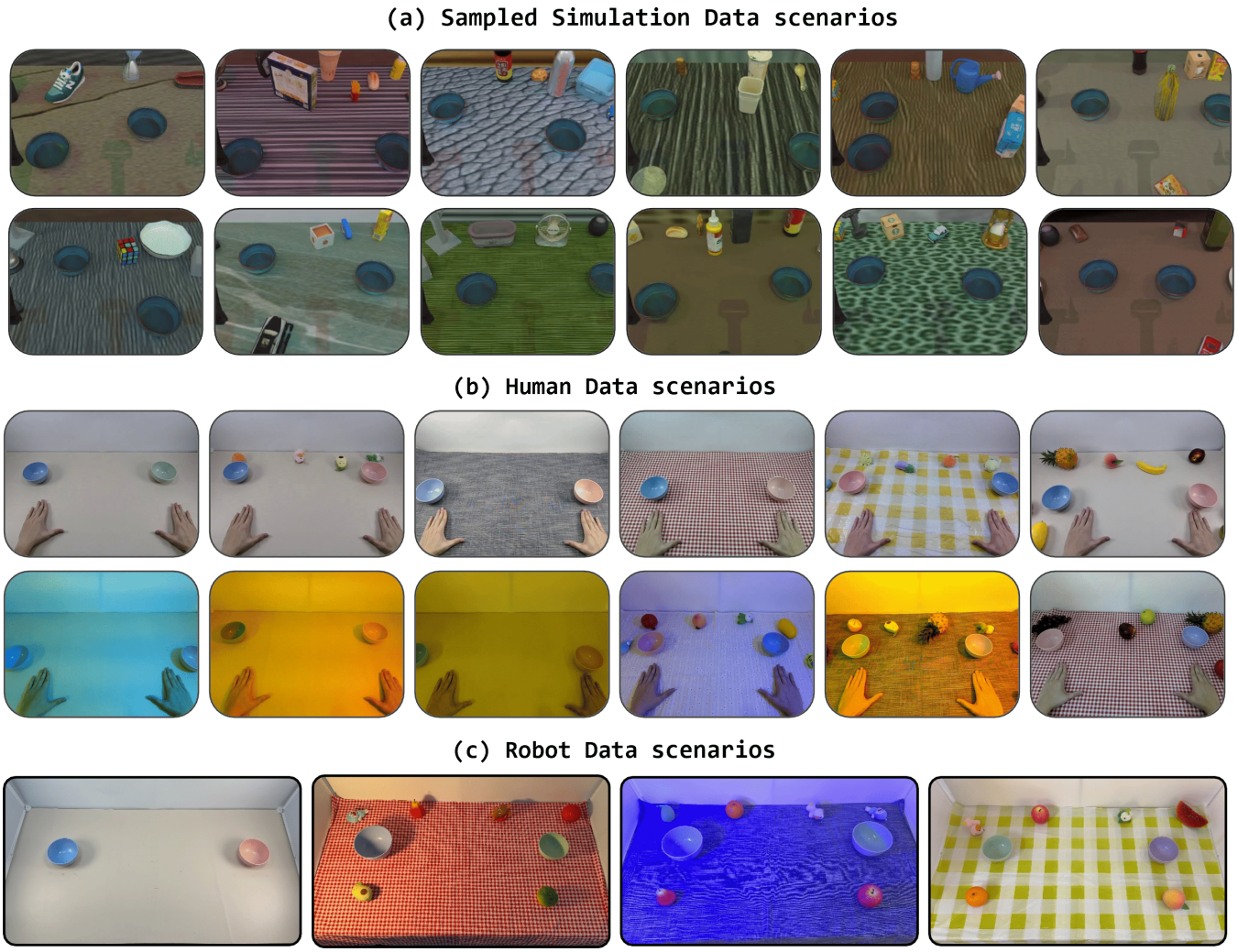

Fig. 13: **Visualization of Data Collection Scenarios.** These diverse environments are constructed by systematically varying visual distractors ($\mathcal{F}_{dist}$), illumination ($\mathcal{F}_{light}$), and background ($\mathcal{F}_{bg}$). (a) Simulation environments leveraging extensive Domain Randomization to synthesize visual diversity. (b) Diverse human demonstration setups constructed by varying these environmental factors. (c) Real-robot setups ranging from simple, clean base scenarios to complex environments with significant clutter and lighting variations.

TABLE III: **Data configurations for the Time Budget Scaling experiment.** The table lists the specific number of episodes collected by each method across different time budgets. The notations **R**, **S**, and **H** denote Real-world robot data, Simulation data, and Human data, respectively.

Method	2h of data	4h of data	8h of data
Real only	40R	80R	160R
HumReal	19R+100H	40R+200H	80R+400H
SimReal	19R+100S	40R+200S	80R+80R+400S
SimHum	19R+100S+100H	40R+200S+200H	80R+400S+400H

I. Implementation Details

We implemented our framework using the PyTorch library and managed training configurations via Hydra. All experiments, including simulation data generation and policy training, were conducted on a workstation equipped with a single NVIDIA RTX 4090 GPU.

Visual Encoder Architecture. To process visual observations, we employed a ResNet-18 backbone initialized with ImageNet-1K pre-trained weights. To capture distinct spatial features, we employ independent visual encoders for each camera view without weight sharing. The extracted feature maps are flattened, concatenated, and projected into the transformer's embedding dimension.

Diffusion Policy Architecture. Our policy architecture employs a shared DiT Backbone consistent with [80]. The diffusion process follows the Denoising Diffusion Probabilistic Models (DDPM), utilizing a squared cosine beta schedule with 100 diffusion steps during training, which is accelerated to 8 steps during inference. To enable unified training on heterogeneous data, we introduce Source-dependent Modules designed to capture the distinct characteristics of each domain. The implementation details of these modules are as follows:

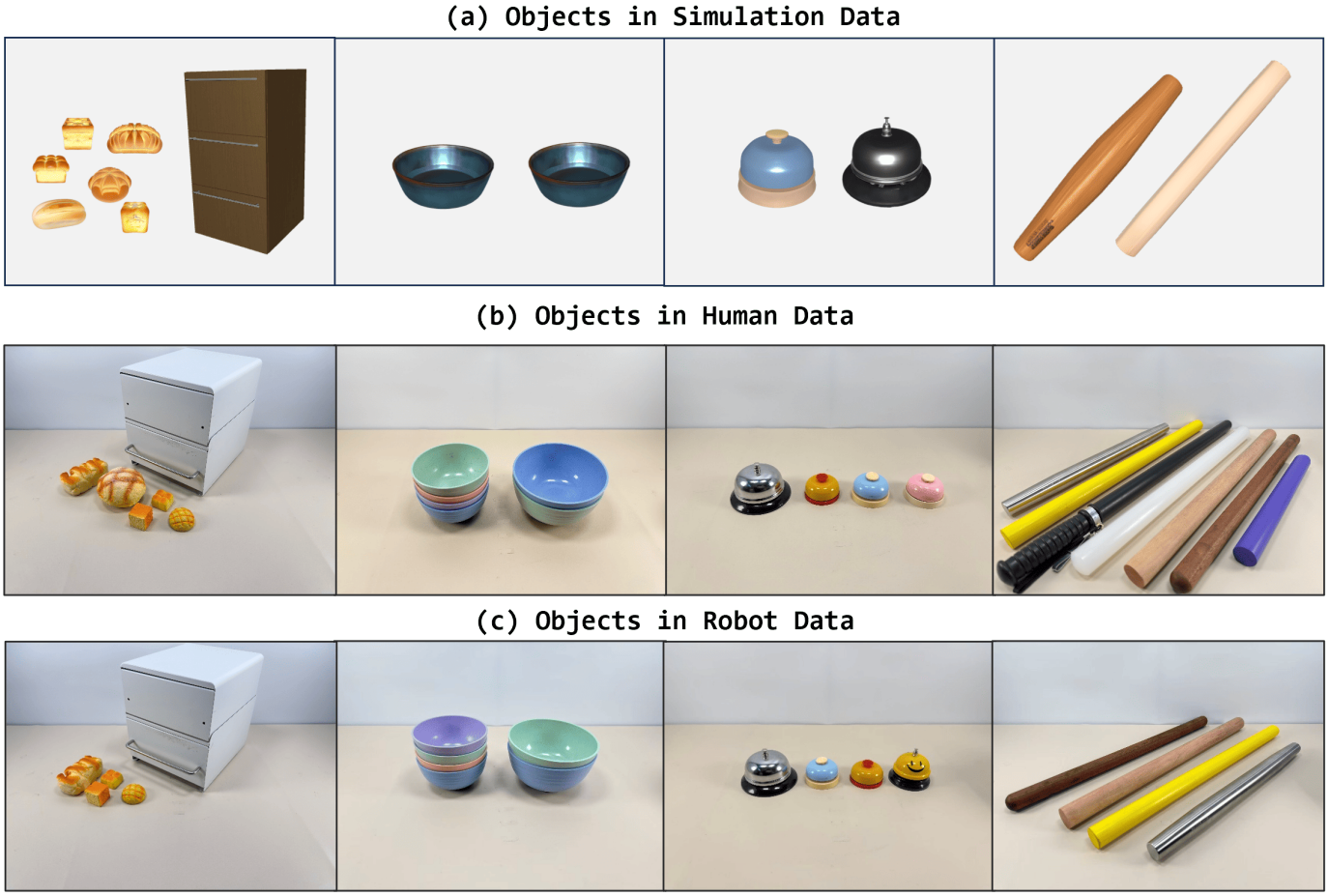

Fig. 14: **Objects used in data collection.** Visualization of the specific object instances ($\mathcal{F}_{obj}$) utilized across different tasks. (a) Synthetic assets from the simulation dataset. (b)Real-world objects from the human dataset. (c) Real-world objects from the real-robot dataset.

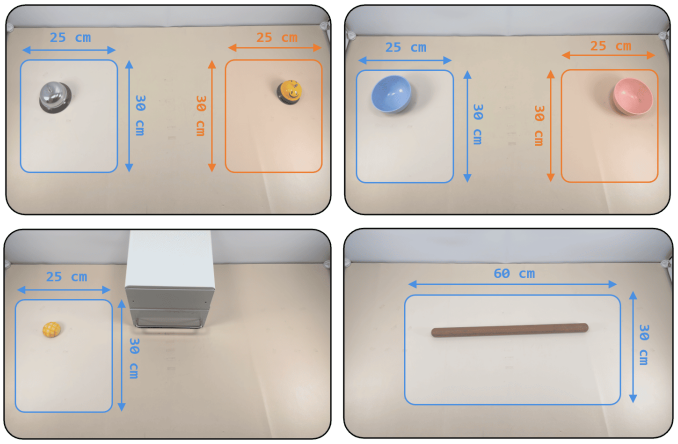

Fig. 15: **Object Initialization Ranges.** Illustration of the spatial coverage ranges used for randomizing object placement ($\mathcal{F}_{init}$) across the four manipulation tasks. The bounding boxes indicate the valid sampling zones (with dimensions in cm) where objects are initialized during both data collection and evaluation trials.

- **Domain-specific Vision Adaptors:** To disentangle the distinct visual information between the real world and simulation, we employ separate vision adaptors. Both adaptors are two-layer MLPs with GELU activations that project high-dimensional visual features into the latent space. Specifically, we utilize a Real-World Adaptor to process the embeddings from human ego-centric videos. In contrast, for the simulation domain, we instantiate independent Simulation Adaptors for each camera view to handle the multi-view synthetic data.
- **Modular Action Encoder:** We utilize separate projection layers to align heterogeneous kinematic spaces before they enter the shared backbone. Proprioceptive State Encoders project the robot state (16-dim) and human state (44-dim) to the hidden dimension using a linear layer preceded by Dropout ($p = 0.2$) to mitigate overfitting. Action Projectors encode the noisy action inputs during diffusion using a non-linear MLP (Linear $\rightarrow$ GELU $\rightarrow$ Linear) to effectively map the distinct action manifolds of human hands and robotic grippers into the shared latent space.
- **Modular Action Decoder:** The latent embeddings from

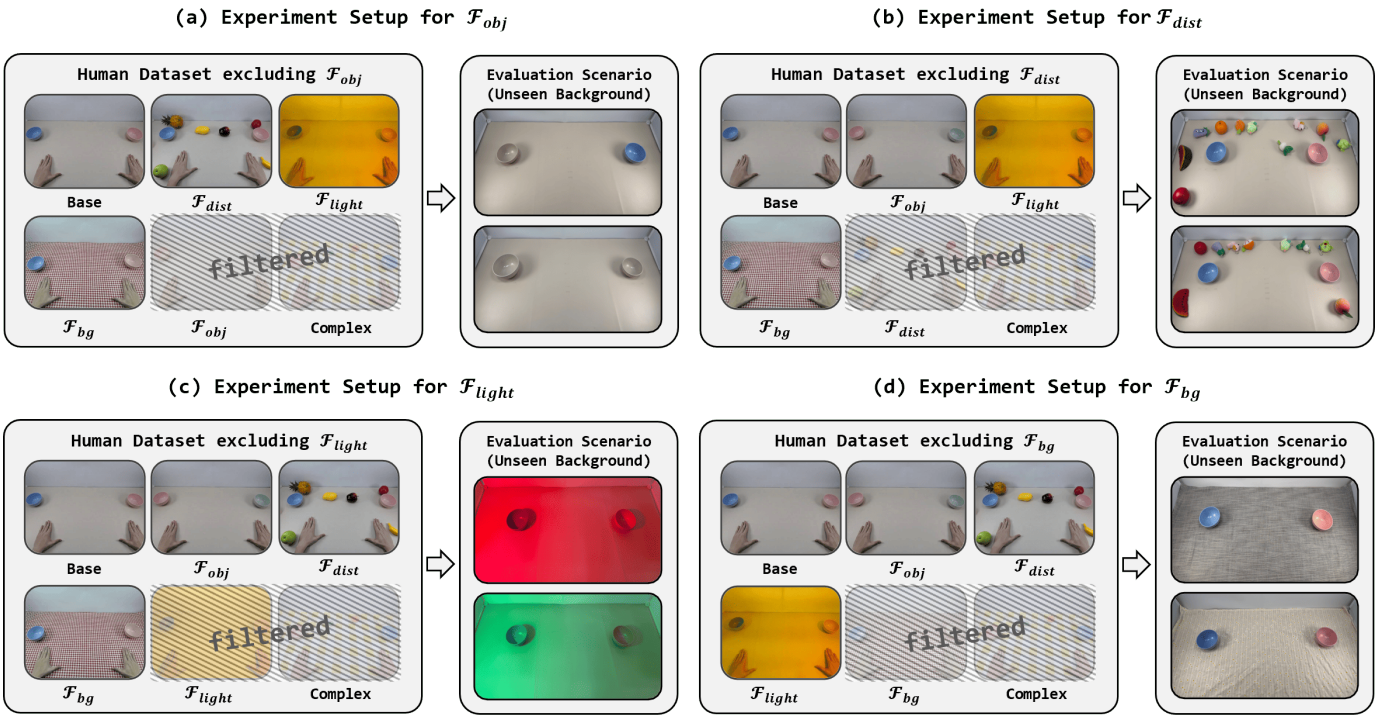

Fig. 16: **Experiment Setup for Environmental Factor Ablation.** We systematically isolate the contribution of specific factors by curating training subsets that exclude data variations corresponding to a target factor (e.g., removing all tablecloth data for $\mathcal{F}_{bg}$). We then evaluate the model in targeted OOD scenarios that explicitly require robustness to the omitted factor, comparing performance against the full human dataset.

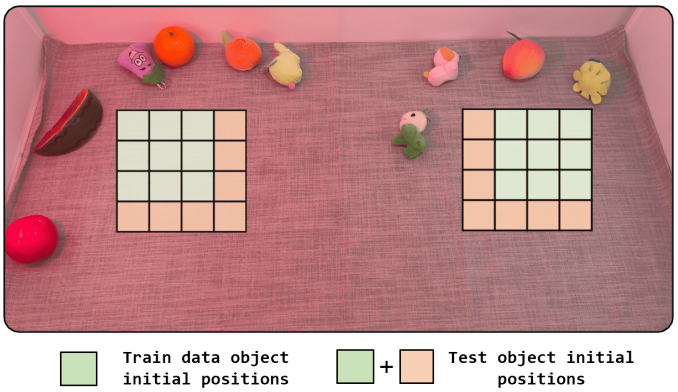

Fig. 17: **Position Generalization Protocol.** Illustration of the discretized 4×4 evaluation grids used in the Stack Bowls Two task. Real-robot training data is strictly confined to the inner 3×3 zones (Green), while evaluation tests the policy's ability to extrapolate to the unseen outer positions (Orange).

the final backbone layer are projected into action space via separate heads. Specifically, distinct linear layers transform the shared features into 16-dimensional robot action noise and 44-dimensional human action noise, respectively.

J. Training Hyperparameters

We outline the specific hyperparameters used for the pre-training and fine-tuning in Table IV.

TABLE IV: Optimization Hyperparameters

Hyperparameter	Pre-training	Fine-tuning
Training Steps	200,000	60,000
Optimizer	AdamW	AdamW
Learning Rate	1×10^{-4}	5×10^{-5}
Optimizer Betas	(0.95, 0.999)	(0.95, 0.999)
Weight Decay	1×10^{-6}	1×10^{-6}
LR Scheduler	Cosine Decay	Cosine Decay
Warmup Steps	2000	2000

K. Author Contributions

- **Algorithm Design:** Kaipeng Fang, Ji Zhang
- **Hardware Setup:** Weiqing Liang, Kaipeng Fang
- **Real-world Evaluation:** Yuyang Li, Weiqing Liang, Kaipeng Fang
- **Data Collection and Processing:** Weiqing Liang, Yuyang Li, Kaipeng Fang
- **Writing and Illustration:** Kaipeng Fang, Ji Zhang, Pengpeng Zeng, Lianli Gao
- **Demo & Website Page:** Weiqing Liang, Kaipeng Fang
- **Infrastructure Support:** Pengpeng Zeng, Jingkuan Song, Heng Tao Shen
- **Project Supervision:** Kaipeng Fang, Lianli Gao, Jingkuan Song, Heng Tao Shen