PhysHSI: Towards a Real-World Generalizable and Natural Humanoid-Scene Interaction System

Huayi Wang[1,*] Wentao Zhang[1,*] Runyi Yu[1,2,*] Tao Huang[1] Junli Ren[1] Feiyu Jia[1] Zirui Wang[1]
Xiaojie Niu[1] Xiao Chen[1] Jiahe Chen[1] Qifeng Chen[2,†] Jingbo Wang[1,†] Jiangmiao Pang[1,†]

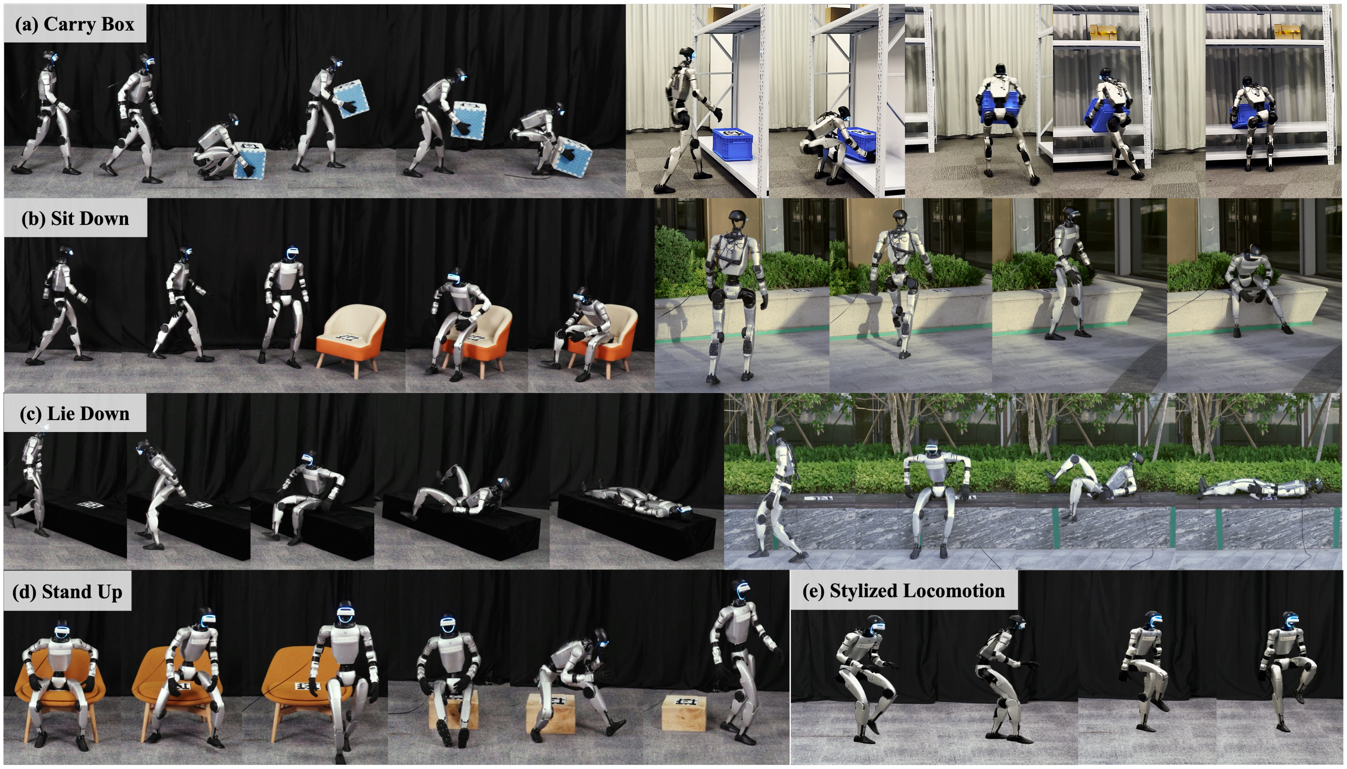

Fig. 1: Our system PhysHSI enables humanoid robots to perform diverse real-world interactions indoors and outdoors with natural behaviors: (a) Carry Box, (b) Sit Down, (c) Lie Down, and (d) Stand Up. PhysHSI can also learn (e) stylized locomotion, such as dinosaur-like walking and high-knee stepping.

***Abstract*— Deploying humanoid robots to interact with real-world environments—such as carrying objects or sitting on chairs—requires generalizable, lifelike motions and robust scene perception. Although prior approaches have advanced each capability individually, combining them in a unified system is still an ongoing challenge. In this work, we present a physical-world humanoid-scene interaction system, PhysHSI, that enables humanoids to autonomously perform diverse interaction tasks while maintaining natural and lifelike behaviors. PhysHSI comprises a simulation training pipeline and a real-world deployment system. In simulation, we adopt adversarial motion prior-based policy learning to imitate natural humanoid-scene interaction data across diverse scenarios, achieving both generalization and lifelike behaviors. For real-world deployment, we introduce a coarse-to-fine object localization module that combines LiDAR and camera inputs to provide continuous and robust scene perception. We validate PhysHSI on four representative interactive tasks—box carrying, sitting, lying, and standing up—in both simulation and real-world settings, demonstrating consistently high success rates, strong generalization across diverse task goals, and natural motion patterns.**

I. Introduction

Imagine deploying humanoid robots in everyday environments—carrying boxes into diverse places or sitting naturally on a chair. Building such a humanoid-scene interaction (HSI) system is considered more sophisticated than executing whole-body skills such as standing up [1, 2], dancing [3, 4], or performing agile motions [5–7]. Beyond these motor capabilities, a real-world HSI system is expected to (1) generalize across diverse interaction scenarios and goals, (2) produce physically plausible and lifelike motions, and (3) incorporate a robust perception module that provides reliable information about surrounding objects and scenes [8].

Existing approaches fall short of these challenges. While classical model-based methods generate stable motions via motion planning or trajectory optimization for tracking [9–13], their high computational cost and strong model assumptions limit generalization to diverse real-world interactions. In contrast, reinforcement learning (RL)-based meth-

[1] Shanghai AI Laboratory, [2] HKUST
* Equal Contributions, † Corresponding Authors
Paper website: https://why618188.github.io/physhsi

ods achieve broader generalization by training from diverse simulated experiences. However, learning policies directly from scratch—whether through a single monolithic policy [14–16] or multiple specialized ones [17–21]—typically requires heavy reward shaping and state transition design, particularly when natural and lifelike motions are desired. To alleviate the hand-crafted design burden and improve motion realism, methods that imitate motion capture (MoCap) priors have been introduced—these approaches effectively yield physically plausible, human-like motions and have driven progress in physics-based character animation for dynamic interactions [22–28]. However, such approaches largely remain confined to simulation and rely on perfect scene observations, leaving sim-to-real transfer an unexplored obstacle.

In this work, we address these challenges by introducing PhysHSI, a real-world system that enables humanoid robots to autonomously perform HSI skills with natural behaviors across diverse goals and interaction scenarios. The system consists of a simulation training pipeline and a real-world deployment module. In simulation, to learn high-quality humanoid interactions, we first curate retargeted MoCap datasets [29, 30] and augment them with manually annotated object information. Using these enriched datasets, we then train generalizable HSI policies via reinforcement learning with adversarial motion priors (AMP) [31, 32], leveraging diverse simulation setups to achieve both natural motion and robust generalization. For real-world deployment, where reliable object localization is challenging due to limited fields of view and frequent occlusions, we design a coarse-to-fine perception module that integrates LiDAR-based odometry for long-range directional cues with camera-based object localization for precise pose estimation at close range.

We evaluate PhysHSI on four representative HSI tasks—box carrying and relocation, sitting on chairs, lying on beds, and standing up from chairs [23]—using Unitree G1 humanoid robots in both simulation and real-world environments. The results show that PhysHSI not only achieves high success rates on these long-horizon tasks but also generalizes effectively across diverse scenarios and task goals. In addition, we demonstrate that PhysHSI produces natural and expressive motions through several learned stylized locomotion behaviors [33]. An overview of the system's real-world performance is provided in Fig. 1.

In summary, our main contribution is introduing PhysHSI, a real-world HSI system that encompasses: (1) an AMP-based training pipeline in simulation that learns from humanoid interaction data, enabling natural and generalizable motions; (2) a coarse-to-fine real-world object localization module that provides continuous and robust scene perception; and (3) evaluation protocols that comprehensively analyze the system and its components, aiming to guide future research and development in real-world HSI tasks.

II. Related Works

A. Humanoid-Scene Interactions

Many works have studied humanoid-scene interaction (HSI) in physics-based simulations, enabling natural, long-horizon behaviors such as object loco-manipulation [23, 26–28, 34]. However, these methods typically rely on idealized task observations and thus face large sim-to-real gaps. For real-world robots, classical approaches often employ model-based motion planning to generate whole-body references for tracking [9–13], but these methods exhibit limited generalization in real-world scenarios. In contrast, RL-based methods learn control policies from scratch with strong generalization by carefully designing rewards and state transitions [14, 16, 17]. To achieve more natural motion, some works leverage curated motion priors to guide policy learning for tasks such as stair climbing and chair sitting [35, 36]. Building on this line of work, our system learns from motion priors to enable generalizable and natural behaviors for more complex interactions, including box carrying and lying down.

B. Humanoid Motion Imitation

Humanoid motion imitation seeks to learn lifelike behaviors from human demonstrations, with motion tracking as a central approach. In simulation, physics-based methods achieve expressive whole-body motions by imitating individual reference sequences [37–39] or learning universal tracking [40]. Recent works extend these methods to real-world robots [3–5, 7], but remain reference-dependent and show limited generalization, constraining interactions with diverse scenes. Adversarial Motion Priors (AMP) [31] improve generalization by imitating motion styles and have been widely studied in simulation [23, 27, 28]. However, real-world applications are limited, with most works using AMP primarily to regularize tracking policies for basic locomotion skills [36, 41–43]. Building on AMP, our system overcomes these limitations and enables natural behaviors for diverse real-world scene and object interactions.

C. Scene Perception

Perception is a fundamental component for enabling humanoid robots to interact with real-world scenes and objects. Motion capture (MoCap) systems can provide accurate global information, supporting highly dynamic interactive tasks [44–46]. However, MoCap is restricted to laboratory environments with limited workspace. To enable more practical deployment, many studies rely on onboard RGB and depth cameras for scene and object perception [15, 47–52]. Yet, these approaches generally confine target objects to a local workspace and often lose sight of them during long-horizon loco-manipulation tasks. Other studies employ LiDAR-Inertial Odometry (LIO) [53, 54] to obtain global information [35, 55–58], though interaction accuracy with objects remains limited. In this work, we propose a coarse-to-fine object localization system that relies solely on onboard sensors and provides continuous and robust scene perception.

III. Simulation Training Pipeline

A. Data Preparation

We begin by preparing humanoid motion data that includes object interactions. While prior works have successfully

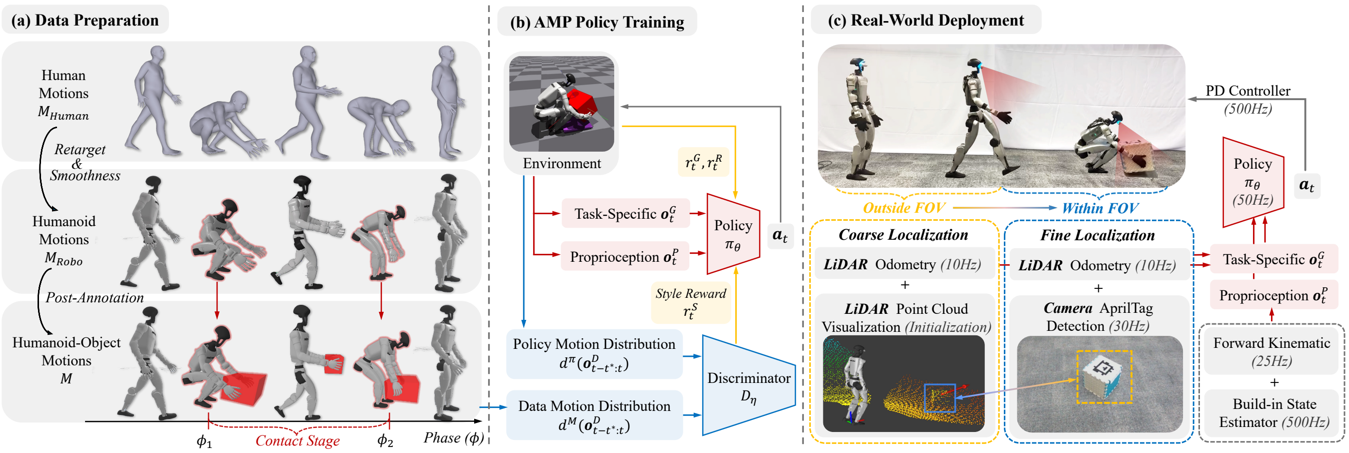

Fig. 2: Overview of PhysHSI. (a) **Dataset Preparation:** Human motions from a MoCap dataset are retargeted to humanoid motions, and objects are annotated by identifying key contact frames. (b) **AMP Policy Training:** A discriminator distinguishes between policy-generated and reference motions to facilitate learning of natural behaviors and task completion. (c) **Real-World Deployment:** The coarse object position is manually specified using LiDAR visualization, and combined with odometry for coarse localization when the object is outside the camera's FOV. Once within view, AprilTag detection combined with odometry is used for fine-grained, automated localization.

retargeted human-only motions onto humanoid robots via optimization [5, 59], generating physically plausible humanoid–object interaction data is more challenging, as it requires maintaining realistic contacts, such as a secure grasp on a box during lifting.

To address this, we adopt a post-annotation strategy for object information. Specifically, we first retarget SMPL motions from the AMASS and SAMP datasets [29, 30] onto the humanoid robot via optimization, applying a smoothing filter to suppress retargeting jitter, yielding a robot-motion-only dataset M_{Robo}. We then manually annotate key contact frames and infer corresponding object trajectories using a simple rule-based procedure: between pickup (ϕ_1) and placement (ϕ_2), the object position $\mathbf{p}^{o_t} \in \mathbb{R}^3$ is set to the midpoint of the hands, with orientation aligned to the robot base; before ϕ_1 and after ϕ_2, it remains fixed at the respective key contact frame. This process produces an augmented humanoid motion dataset M with consistent and physically coherent object positions, which is crucial for stage conditioning and reference state initialization (Sec. III-B).

B. Adversarial Motion Prior Policy Training

We formulate the humanoid-scene interaction (HSI) problem as a reinforcement learning (RL) task. To enable humanoids to interact with objects in a lifelike manner while generalizing across diverse scenarios, we build on the Adversarial Motion Priors (AMP) framework [31], which has two components: a policy π_θ that generates humanoid actions, and a discriminator $\mathcal{D}$ that distinguishes between policy motions and those in the reference motion dataset.

1) Observation and Action Space: The policy observation $\mathbf{o}_t^\pi$ at each timestep t consists of a 5-step history of proprioception $\mathbf{o}_{t-4:t}^P$ and task-specific observations $\mathbf{o}_{t-4:t}^G$.

The proprioception $\mathbf{o}_t^P \in \mathbb{R}^{108}$ is defined as

$$\mathbf{o}_t^P \triangleq \left[\boldsymbol{\omega}_{b_t}, \mathbf{g}_{b_t}, \boldsymbol{\theta}_t, \dot{\boldsymbol{\theta}}_t, \mathbf{p}_{b_t}^{ee}, \mathbf{a}_{t-1}\right], \tag{1}$$

where b_t represent robot base frame t, $\boldsymbol{\omega}_{b_t} \in \mathbb{R}^3$ is the base angular velocity, $\mathbf{g}_{b_t} \in \mathbb{R}^3$ is the base gravity direction, $\boldsymbol{\theta}_t \in \mathbb{R}^{29}$ and $\dot{\boldsymbol{\theta}}_t \in \mathbb{R}^{29}$ are joint positions and velocities respectively, $\mathbf{p}_{b_t}^{ee} \in \mathbb{R}^{5\times3}$ denotes the 3D positions of five end-effectors (left/right hand/foot, and head) in the base frame, and $\mathbf{a}_{t-1}$ is the action taken at the previous timestep.

The task-specific observation $\mathbf{o}_t^G$ varies depending on the task. In general, it consists of three components: (a) the object shape $\mathbf{b}_t \in \mathbb{R}^3$, represented by its bounding box dimensions; (b) the object position $\mathbf{p}_{b_t}^{o_t} \in \mathbb{R}^3$ and orientation $\mathbf{R}_{b_t}^{o_t} \in \mathbb{R}^6$ encoded with a 6D normal-tangent representation [60]; and (c) the target goal position $\mathbf{p}_{b_t}^{g_t} \in \mathbb{R}^3$. All quantities are expressed in the robot's base frame.

The discriminator observation $\mathbf{o}_t^{\mathcal{D}} \in \mathbb{R}^{57}$ at each timestep consists of privileged information and is defined as

$$\mathbf{o}_t^{\mathcal{D}} \triangleq \left[h_t, \mathbf{v}_{b_t}, \boldsymbol{\omega}_{b_t}, \mathbf{g}_{b_t}, \boldsymbol{\theta}_t, \mathbf{p}_{b_t}^{ee}, \mathbf{p}_{b_t}^{o_t}\right], \tag{2}$$

where $h_t \in \mathbb{R}$ denotes the base height, $\mathbf{v}_{b_t} \in \mathbb{R}^3$ is the base linear velocity, $\boldsymbol{\omega}_{b_t}$ is the base angular velocity. Notably, including the object position $\mathbf{p}_{b_t}^{o_t}$ in the discriminator observation is crucial for long-horizon tasks, as it lets the discriminator implicitly condition on task phases—approach, pickup, carry, or place—enhancing policy training guidance.

The action $\mathbf{a}_t \in \mathbb{R}^{29}$ from policy $\pi_\theta(\mathbf{o}_t^\pi)$ specifies target joint positions, executed by a PD controller across all 29 humanoid DoFs.

2) Reward Terms and Discriminator Learning: The reward function is defined as the sum of three components: $r_t \triangleq w^G r_t^G + w^R r_t^R + w^S r_t^S$, where r_t^G is the task reward encouraging the humanoid to achieve high-level objectives, r_t^R regularizes excessive joint torques and joint speed, r_t^S is the style reward that encourages the humanoid to imitate behaviors from the reference motion dataset, and $w^{(\cdot)}$ denotes the corresponding coefficients.

The style reward is modeled using the adversarial discriminator $\mathcal{D}$, trained to differentiate between motions produced by the policy and those in the dataset. The discriminator is optimized according to [31]:

$$\begin{aligned}\arg\min_{\mathcal{D}} & -\mathbb{E}_{d^{M}(\mathbf{o}^{\mathcal{D}}_{t:t+t^*})}\left[\log\left(\mathcal{D}(\mathbf{o}^{\mathcal{D}}_{t:t+t^*})\right)\right] \\ & -\mathbb{E}_{d^{\pi}(\mathbf{o}^{\mathcal{D}}_{t:t+t^*})}\left[\log\left(1-\mathcal{D}(\mathbf{o}^{\mathcal{D}}_{t:t+t^*})\right)\right] \\ & + w^{\mathrm{gp}}\mathbb{E}_{d^{M}(\mathbf{o}^{\mathcal{D}}_{t:t+t^*})}\left[\left\| \nabla_{\eta}\mathcal{D}(\eta)|_{\eta=(\mathbf{o}^{\mathcal{D}}_{t:t+t^*})}\right\|^2\right],\end{aligned} \tag{3}$$

where $d^M(\mathbf{o}^{\mathcal{D}}_{t:t+t^*})$ and $d^{\pi}(\mathbf{o}^{\mathcal{D}}_{t:t+t^*})$ denote the distributions of (t^*+1)-frame motion clips from the dataset M and the policy π_θ, respectively, and w^{gp} is a coefficient that regularizes the gradient penalty [61] in adversarial training. Finally, the style reward for the policy is specified as

$$r_t^S \triangleq -\log\left(1-\mathcal{D}(\mathbf{o}^{\mathcal{D}}_{t-t^*:t})\right). \tag{4}$$

To optimize the policy, we use the proximal policy optimization (PPO) [62] to maximize the cumulative discounted reward $\mathbb{E}\left[\sum_{t=1}^{T}\gamma^{t-1}r_t\right]$.

3) Hybrid Reference State Initialization: Many HSI tasks are long-horizon, and directly initializing all episodes from the default starting pose makes exploration difficult, since the humanoid rarely experiences critical transitions. To address this, we adopt the reference state initialization (RSI) strategy [37], which initializes episodes from randomly sampled reference motions along with the corresponding labeled object states, thereby improving exploration efficiency.

This naive RSI strategy, however, risks overfitting to the limited scene configurations in the demonstrations. We mitigate this limitation in two ways. First, we leverage the compositional nature of task stages: while a motion clip may specify the pickup position of the box, the subsequent goal position does not need to match the data. Thus, we sample an initial phase $\phi \in [0,1]$ from motion data, while randomizing the scene for $(\phi, 1]$. Second, a subset of episodes are initialized from the default starting pose with fully randomized scene parameters (e.g., object size, position, and goal position). This hybrid RSI strategy promotes efficient exploration while ensuring generalization.

4) Asymmetric Actor-Critic Training: In real-world, the agent receives only partial observations due to noise and sensing limitations. System constraints further require masking some task observations during training (see Sec. IV-B). To compensate, we adopt the asymmetric actor-critic framework [63], where the actor uses inputs $\mathbf{o}_t^{\pi}$ available at deployment, while the critic observes a richer state $\mathbf{o}_t^{V}$ (e.g., base velocity and unmasked task observations).

5) Motion Constraints: As rewards accumulate across stages, the agent tends to exploit shortcuts by producing fast, jerky motions, especially later in training, which are unsuitable for deployment. To address this, we assign a small style reward weight w^S early for exploration, and gradually increase it to align with motion data. Additionally, we adopt the L2C2 smoothness regularization [64] to enhance smoothness and stability for hardware deployment.

IV. Real-World Deployment System

To deploy the trained HSI skills in the real world, two key observations must be obtained: the end-effector position $\mathbf{p}^{ee}_{b_t}$ and the object pose—position $\mathbf{p}^{o_t}_{b_t}$ and orientation $\mathbf{R}^{o_t}_{b_t}$—in the robot base frame at time t. It is easy to get accurate $\mathbf{p}^{ee}_{b_t}$ by forward kinematics (FK) with joint encoder information. In contrast, reliable object localization is more challenging, as onboard sensors often suffer from limited fields of view and frequent occlusions—for instance, when the robot starts with no object visible or when the object moves out of view during motion. To overcome these and obtain robust, continuous localization, we design a coarse-to-fine perception system (Sec. IV-A) that integrates LiDAR and RGB camera inputs. We further adapt the simulation training to align with this perception pipeline (Sec. IV-B) and describe the corresponding hardware setup in Sec. IV-C.

A. Coarse-to-Fine Object Localization

We represent position and orientation using the transform matrix for clarity. Specifically,

$$T^{o_t}_{b_t} = f_{\mathrm{T}}(\mathbf{p}^{o_t}_{b_t}, \mathbf{R}^{o_t}_{b_t}) \in SE(3) \tag{5}$$

denotes the pose of object o at time t in the robot frame b_t, where $f_{\mathrm{T}}(\cdot)$ maps position $\mathbf{p}^{o_t}_{b_t}$ and orientation $\mathbf{R}^{o_t}_{b_t}$ to the transform matrix. At initialization, the target object is often outside the camera's field of view. We therefore assign a coarse initial pose $T^{o_0}_{b_0}$, where the position $\mathbf{p}^{o_0}_{b_0}$ is manually specified using LiDAR point cloud visualization, and the orientation $\mathbf{R}^{o_0}_{b_0}$ is set as default from identity rotation matrix.

During execution, when the robot is far from the object, we use FAST-LIO [53] to estimate the odometry $T^{b_t}_{b_0}$, i.e., the pose of the current base frame with respect to the initial frame. The object position in the current base frame is then obtained as:

$$\mathbf{p}^{o_t}_{b_t}, \mathbf{R}^{o_t}_{b_t} = f_{\mathrm{T}}^{-1}((T^{b_t}_{b_0})^{-1}T^{o_0}_{b_0}), \tag{6}$$

where $f_{\mathrm{T}}^{-1}(\cdot)$ extracts position and orientation from a transformation matrix. This provides a continuous but coarse estimate of the object pose, sufficient to guide the robot toward the target from long range.

For fine-grained localization at close range, AprilTag detection [65] is employed to provide accurate object position $\mathbf{p}^{o_t}_{c_t}$ and orientation $\mathbf{R}^{o_t}_{c_t}$ in the camera frame c_t. Coarse localization automatically transitions to fine localization upon the tag's first detection. Temporary detection losses (e.g., when the robot turns to sit down) are handled by retaining the last observed object pose $T^{o_{t'}}_{c_{t'}}$ and corresponding FK information $T^{c_{t'}}_{b_{t'}}$, which are then propagated to the current time t using odometry $T^{b_t}_{b_{t'}}$, following the same principle as Eq. 6:

$$\mathbf{p}^{o_t}_{b_t}, \mathbf{R}^{o_t}_{b_t} = f_{\mathrm{T}}^{-1}\left((T^{b_t}_{b_{t'}})^{-1}T^{c_{t'}}_{b_{t'}}T^{o_{t'}}_{c_{t'}}\right). \tag{7}$$

We further distinguish between static and dynamic objects. For static objects (e.g., chairs), the pose is assumed fixed and updated via the propagation strategy described above, such as when the robot prepares to turn and sit down. For dynamic objects (e.g., boxes), this estimation is valid until grasping; after grasping, if the object leaves the camera view, both position and orientation are masked, and proprioception is relied to complete the task. A simple distance threshold ϵ defines the grasp phase: if the estimated object distance

TABLE I: Benchmarked Comparison in Simulation.

	Carry Box		*Sit Down*		*Lie Down*		*Stand Up*	
	$R_{\text{succ}}(\%,\uparrow)$	$S_{\text{human}}(\uparrow)$	$R_{\text{succ}}(\%,\uparrow)$	$S_{\text{human}}(\uparrow)$	$R_{\text{succ}}(\%,\uparrow)$	$S_{\text{human}}(\uparrow)$	$R_{\text{succ}}(\%,\uparrow)$	$S_{\text{human}}(\uparrow)$
In Distribution Scene								
RL-Rewards	72.92 (±8.29)	1.67 (±0.47)	83.60 (±5.98)	1.50 (±0.24)	76.72 (±9.43)	0.50 (±0.00)	93.02 (±0.71)	1.50 (±0.24)
Tracking-Based	11.84 (±3.16)	**4.83** (±0.24)	31.46 (±2.96)	3.80 (±0.08)	19.58 (±1.02)	2.23 (±0.21)	99.00 (±1.28)	**4.67** (±0.12)
PhysHSI	**91.34** (±1.63)	4.00 (±0.41)	**96.28** (±0.21)	**4.80** (±0.08)	**97.86** (±0.60)	**4.80** (±0.08)	**99.68** (±0.21)	3.77 (±0.21)
Full Distribution Scene								
RL-Rewards	63.40 (±8.63)	1.17 (±0.24)	73.14 (±4.29)	3.07 (±0.09)	55.76 (±12.51)	2.00 (±1.08)	90.50 (±2.33)	1.07 (±0.09)
Tracking-Based	0.02 (±0.01)	0.50 (±0.00)	1.12 (±0.51)	0.50 (±0.00)	0.94 (±0.45)	1.00 (±0.41)	35.32 (±2.51)	3.27 (±0.54)
PhysHSI	**84.60** (±3.74)	**3.83** (±0.24)	**91.32** (±2.48)	**4.77** (±0.05)	**81.28** (±3.99)	**4.43** (±0.33)	**92.24** (±0.75)	**3.77** (±0.52)

exceeds ϵ, the object is treated as static; otherwise, it is assumed to move with the robot.

B. Sim-to-Real Transfer

To better match real-world observations, we apply domain randomization [66]. Two key strategies are used: (1) adding random offsets, Gaussian noise, and delays to object poses and FK observations; (2) replicating the masking mechanism for dynamic objects during the grasping stage, which is applied when the object is outside the camera's view, the goal distance is out of range, or the camera angle deviates excessively from vertical. We further adopt standard domain randomization techniques from [55] to enhance robustness and facilitate sim-to-real transfer.

C. Hardware Setup

Our system is built on the Unitree G1 humanoid robot, equipped with a built-in Livox Mid-360 LiDAR and an external Intel RealSense D455 depth camera mounted on the head, with a 86° horizontal and 57° vertical field of view. Perception modules—including point cloud visualization, Fast-LIO, AprilTag detection, and forward kinematics—together with the learned policy, all run onboard on a Jetson Orin NX, enabling fully portable deployment.

V. Simulation Experiments

In this section, we validate the effectiveness of our simulation training pipeline and conduct ablation studies to assess the contribution of each module.

A. Experimental Setup

We compare PhysHSI to two commonly adopted baselines:

- **RL-Rewards**: The humanoid learns HSI tasks from scratch without motion references, using a combination of gait, task, and regularization RL rewards.
- **Tracking-Based**: The agent mimics motion references by tracking humanoid and object trajectories provided by the dataset. We use the same dataset as in PhysHSI, which contains roughly 2–5 complete trajectories per task.

All training and evaluation environments are implemented in IsaacGym [67]. We benchmark methods on four representative HSI tasks: ***carry box*** (walk to, lift, carry, and place the box), ***sit down*** (walk to and sit on a chair), ***lie down*** (walk to and lie on a bed), and ***stand up*** (rise from a chair).

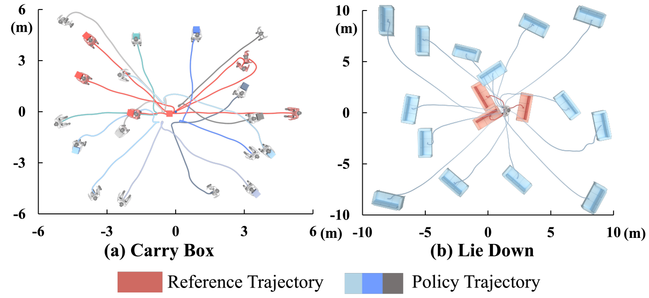

Fig. 3: Spatial Generalization. Root trajectories of the robot are shown for tasks (a) Carry Box and (b) Lie Down. Red trajectories indicate reference data, with others representing sampled policy motions.

For evaluation, we consider two settings: ***in-distribution scenes***, which only include scene settings from the dataset, and ***full-distribution scenes***, where scenes are uniformly sampled within the task space (objects placed within $[0, 5]\,\text{m}$ of the start position; boxes initialized at heights within $[0, 0.6]\,\text{m}$ and size dimensions within $[0.2, 0.5]\,\text{m}$).

We report two metrics: ***success rate*** (R_{succ}) and ***human-likeness score*** (S_{human}). R_{succ} measures whether the object is correctly placed or the humanoid reaches the desired pose. S_{human} is evaluated by Gemini-2.5-Pro [68], which, given task descriptions and experimental trajectories, assigns a human-like score ranging from 0 to 5 for each demonstration clip. For each setting, the mean and standard deviation are computed over five random seeds, each evaluated across 1000 episodes and three demo clips.

B. Overall Performance

PhysHSI achieves high success rates and produces natural, lifelike motions across all tasks, as shown in Table I. Key findings are as follows:

Consistently High Success Rates PhysHSI completes all four long-horizon HSI tasks with consistently strong performance. In the more challenging *carry box* task with four subtasks, it reaches an 81.34% success rate, comparable to the simpler two-step *sit down* task.

Strong Generalization Unlike tracking-based methods that mimic reference trajectories frame by frame, PhysHSI leverages AMP frameworks to enable flexible motion recombination, requiring only style alignment with motion priors. This enables comparable success rates even in full-distribution scenes, whereas tracking-based methods almost completely fail (near 0 success) due to the limited scale of reference data. Fig. 3 shows partially successful trajectories, highlighting the strong generalization learned from only a few references.

Lifelike Motion Patterns PhysHSI attains significantly higher S_{human} than RL reward–based methods. By training the policy and discriminator in competition, our approach effectively distinguishes between dataset motions and policy-generated motions, leading to natural behaviors. In contrast, RL reward methods require carefully hand-crafted gait and regularization terms, which are difficult to design and less effective for long-horizon tasks.

C. Ablation Analysis

We conduct ablation studies on the data processing, RSI strategy and mask strategy. The main results are presented in Table II, with the following key observations:

Data quality and object annotation are critical for natural motion and task completion. Training without smoothed motion data (w/o Smoothness) produces unnatural behaviors, as the policy—guided by the discriminator—may exploit artifacts such as jittering end-effectors or abrupt motion shifts. Removing object annotations (w/o Object) increases failure rates, since object states in AMP observations are essential for learning stage transitions and motion styles. For instance, when the object is distant, the discriminator drives the humanoid to walk toward it, while during carrying, it keeps the box centered between the hands.

Hybrid RSI is crucial for generalization and efficiency. We compare our hybrid RSI with two alternatives: no RSI (w/o RSI) and naive RSI, where all episodes are initialized from reference states fixed to dataset settings. Naive RSI performs worse than no RSI, demonstrating poor generalization and low training efficiency due to the limited diversity of observed scenes. In contrast, hybrid RSI significantly improves both generalization and sample efficiency.

Mask processing has a limited impact on overall performance. Although the masking strategy introduced in Sec. IV-B slightly slows training compared to using complete object states (w/o Obs Mask), which represent an upper bound on performance, it only minimally affects the final policy success rate in the two ablation cases.

VI. Real-World Experiments

In this section, we evaluate the overall system performance in real-world scenarios and assess the effectiveness of our proposed coarse-to-fine object localization module.

A. Overall Performance

As shown in Fig. 1, PhysHSI achieves zero-shot transfer and successfully completes all four HSI tasks in real-world settings. We further evaluate success rate R_{succ}, finish precision $R_{\text{precision}}$, execution time T_{exec}, and maximum movement range M_{range} with 10 trials per task, as reported in Table III. Our key findings are summarized below:

- PhysHSI achieves competitive success rates with high precision in real-world deployments across all four tasks, showing particularly strong performance on *lie down* and *sit down*. For the more challenging *carry box* task, the system attains an 8/10 success rate for lifting and 6/10 for the full sequence, with placement errors under 20 cm.

TABLE II: Ablation Experiments.

	Carry Box		*Sit Down*	
	$R_{\text{succ}}(\%,\uparrow)$	$S_{\text{human}}(\uparrow)$	$R_{\text{succ}}(\%,\uparrow)$	$S_{\text{human}}(\uparrow)$
Ablation on Data Processing				
w/o Smoothness	63.28 (±11.72)	2.33 (±1.03)	87.24 (±2.19)	1.33 (±0.23)
w/o Object	55.42 (±8.17)	2.60 (±0.57)	72.36 (±6.71)	3.50 (±0.70)
PhysHSI	**79.34** (±4.71)	**3.83** (±0.24)	**91.32** (±2.48)	**4.77** (±0.05)
Ablation on RSI Strategy				
w/o RSI	41.24 (±6.92)	2.50 (±1.63)	78.24 (±3.91)	4.50 (±0.00)
Naive RSI	5.70 (±2.38)	0.50 (±0.0)	18.70 (±5.33)	1.83 (±0.62)
Hybrid RSI	**79.34** (±4.71)	**3.83** (±0.24)	**91.32** (±2.48)	**4.77** (±0.05)
Ablation on Mask Strategy *(for dynamic objects)*				
w/o Obs Mask	**85.90** (±2.90)	**4.30** (±0.14)	/	/
PhysHSI	79.34 (±4.71)	3.83 (±0.24)	**91.32** (±2.48)	**4.77** (±0.05)

TABLE III: Real-World Experiments. Success rates for the pick-up stage and the full sequence are separately reported for the *Carry Box* task.

Tasks	R_{succ}	$R_{\text{precision}}$ (m)	T_{exec} (s)	M_{range} (m)
Carry Box	8/10, 6/10	0.19 (±0.10)	10.5 (±2.8)	5.69
Sit Down	9/10	0.07 (±0.03)	6.2 (±1.3)	4.14
Lie Down	8/10	0.16 (±0.07)	6.7 (±1.0)	3.76
Stand Up	8/10	/	2.3 (±0.4)	1.74

- PhysHSI generalizes effectively across variations in spatial layout and object properties, handling locomotion over distances up to 5.7 m with diverse box dimensions, heights, and weights. Representative examples of varied scene configurations are shown in Fig. 4.
- Compared to reward-tuned RL policies, PhysHSI generates more natural, human-like motions. Our policy inherits the catwalk-style locomotion present in AMASS data, while the framework also supports stylized motion learning. As shown in Fig. 1(e), the system can produce diverse locomotion styles, such as dinosaur-like walking or high-knee stepping.
- PhysHSI can be deployed outdoors using only onboard sensing and computation (Fig. 1(a)-(c)). This highlights the portability of our system compared to MoCap-based deployments that rely on external infrastructure.

B. Object Localization Module Analysis

To evaluate the effectiveness of our object localization module, we conducted 17 real-world HSI trials, 15 of which were successful. For each trial, we recorded the object trajectories estimated by our module and compared them against ground-truth trajectories obtained from a MoCap system. We also measured the robot-object distance at the coarse-to-fine transition point. As shown in Fig. 5(a), localization error is relatively large (0.35 m) when the robot is far from the object. Once within 2.4 m, AprilTag detection activates, switching to fine localization with an average error of 0.05 m. These results demonstrate the effectiveness of our design: the coarse stage provides reliable directional guidance at long range, while the fine stage yields accurate positions at close range.

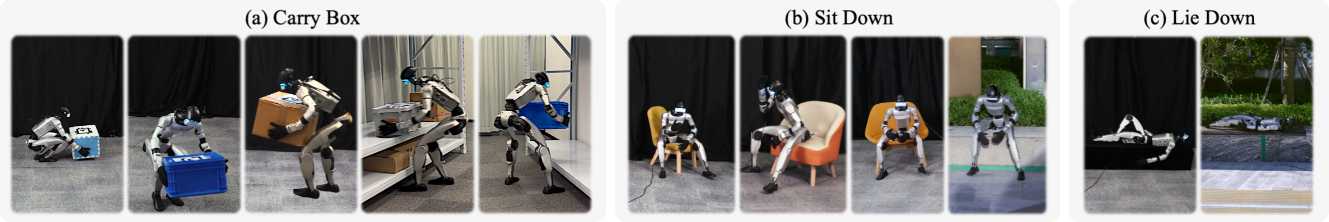

Fig. 4: Real-World Generalization. PhysHSI generalizes to diverse real-world scenes, (a) handling boxes of varying shapes, weights, and heights, and (b) sitting or (c) lying on chairs and beds of different heights, both indoors and outdoors.

TABLE IV: Limitation Test for *Carry Box* Task.

Test Condition	Box Height				Box Weight					Maximum Box Size			
	0 cm	20 cm	40 cm	60 cm	0.6 kg	1.2 kg	2.3 kg	3.6 kg	4.5 kg	20 cm	30 cm	40 cm	45 cm
$R_{\text{succ}}(\uparrow)$	2/3	3/3	3/3	1/3	2/3	3/3	2/3	1/3	0/3	2/3	3/3	2/3	3/3

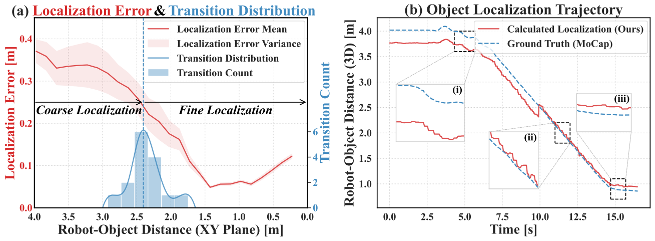

Fig. 5: Real-World Localization System Analysis. (a) Localization error versus robot–object distance, with coarse-to-fine transition statistics and distribution. (b) A representative object localization trajectory, highlighting three stages: (i) coarse localization, (ii) fine localization, and (iii) grasp.

The overall success rate of 15/17 further confirms robustness. Two failures occurred due to coarse guidance deviating too far (preventing tag from entering FOV) and a system crash.

To further analyze error sources across stages, we examine one successful trajectory by comparing the estimated localization with the ground-truth trajectory (Fig. 5(b)). Three stages are observed: (i) **Coarse stage**: errors mainly arise from manually specified goal points in LiDAR point-cloud visualization, deviating $0.3\,\mathrm{m}$ from the exact position. Despite this, the estimated and ground-truth trajectories show consistent trends, which is sufficient for guidance. (ii) **Fine stage**: errors stem from odometry drift and AprilTag noise, but remain small, with trajectories closely aligned. (iii) **Grasping stage**: at close range, errors are dominated by AprilTag noise and are more pronounced due to rapid manipulative motions compared to smoother locomotion.

C. System Limitation Analysis

We analyze the limitations of our system on the *carry box* task. We evaluate different carrying heights, box masses, and shapes, each with three trials, and report the success rates in Table IV. We find that the humanoid can stably carry boxes at heights in $[0, 60]$ cm, with weights in $[0.6, 3.6]$ kg, and maximum sizes up to $[20, 45]$ cm. Carrying higher boxes exceeds the robot's vertical FOV even when stationary, while heavier or wider boxes cannot be handled due to the limited reach of the rubber hand and arm length.

Beyond these findings, we identify several broader limitations that highlight challenges in advancing HSI capabilities:

Hardware Constraints. The current system relies on the Unitree G1's rubber hand for clamping, which restricts manipulation of larger or heavier boxes. Excessive weight may also cause motor overheating and potential hardware failures during execution.

Large-Scale High-Quality HSI Data. In this work, we post-annotate objects in retargeted humanoid motion data and select a small subset of high-quality samples for training. However, this manual process does not scale well when large scale of high-quality HSI data are required.

Automated Perception Module. Our current object localization relies on a modular real-world system, which introduces complexity and potential fragility. Developing a more automated perception module, for example with active perception that enables autonomous exploration, could improve robustness and simplify deployment.

VII. Conclusions

We presented PhysHSI, a real-world system for generalizable and natural humanoid-scene interaction, combining an effective simulation training pipeline with a robust deployment module. PhysHSI successfully performs tasks such as *carry box* and *lie down* in real-world scenarios with high success rates, strong spatial and object-level generalization, and natural motion behaviors. Moreover, with only a manually specified coarse object initialization and a single fiducial tag, our system can autonomously complete tasks even in outdoor environments, demonstrating its portability. This work represents an initial exploration of real-world HSI tasks and paves the way for more advanced object- and scene-interaction capabilities in practical applications.

Acknowledgements

This work is funded in part by the National Key R&D Program of China (2022ZD0160201), and Shanghai Artificial Intelligence Laboratory. We thank Liang Pan for advice on the implementation of RSI. We thank Shunlin Lu for help with the process of motion data. We thank Jianhui Liu, Tai Wang, Qingwei Ben and Junfeng Long for valuable discussions and advice on the object localization module. We thank Chenhui Li and Intelligent Photonics and Electronics Center at Shanghai AI Lab for help with the MoCap system and SLAM devices. We thank Weixiang Zhong and Yinhuai

Wang for assistance with the real-world experiments. We thank Unitree and the Hardware Team of the Embodied AI Center at Shanghai AI Lab for help with hardware issues and the Unitree G1 humanoid robot.

References

[1] T. Huang *et al.*, "Learning humanoid standing-up control across diverse postures," in *Robotics: Science and Systems (RSS)*, 2025.

[2] X. He, R. Dong, Z. Chen, and S. Gupta, "Learning getting-up policies for real-world humanoid robots," in *Robotics: Science and Systems (RSS)*, 2025.

[3] X. Cheng, Y. Ji, J. Chen, R. Yang, G. Yang, and X. Wang, "Expressive whole-body control for humanoid robots," in *Robotics: Science and Systems (RSS)*, 2024.

[4] M. Ji *et al.*, "Exbody2: Advanced expressive humanoid whole-body control," *arXiv preprint arXiv:2412.13196*, 2024.

[5] T. He *et al.*, "Asap: Aligning simulation and real-world physics for learning agile humanoid whole-body skills," in *Robotics: Science and Systems (RSS)*, 2025.

[6] W. Xie *et al.*, "Kungfubot: Physics-based humanoid whole-body control for learning highly-dynamic skills," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

[7] Q. Liao, T. E. Truong, X. Huang, G. Tevet, K. Sreenath, and C. K. Liu, "Beyondmimic: From motion tracking to versatile humanoid control via guided diffusion," *arXiv preprint arXiv:2508.08241*, 2025.

[8] Z. Gu *et al.*, "Humanoid locomotion and manipulation: Current progress and challenges in control, planning, and learning," *Transactions on Mechatronics*, 2025.

[9] F. Ruscelli, M. P. Polverini, A. Laurenzi, E. M. Hoffman, and N. G. Tsagarakis, "A multi-contact motion planning and control strategy for physical interaction tasks using a humanoid robot," in *International Conference on Intelligent Robots and Systems (IROS)*, 2020.

[10] N. Figueroa, S. Faraji, M. Koptev, and A. Billard, "A dynamical system approach for adaptive grasping, navigation and co-manipulation with humanoid robots," in *International Conference on Robotics and Automation (ICRA)*, 2020.

[11] M. Sombolestan and Q. Nguyen, "Hierarchical adaptive control for collaborative manipulation of a rigid object by quadrupedal robots," in *International Conference on Robotics and Automation (ICRA)*, 2023.

[12] A. Adu-Bredu, G. Gibson, and J. Grizzle, "Exploring kinodynamic fabrics for reactive whole-body control of underactuated humanoid robots," in *International Conference on Intelligent Robots and Systems (IROS)*, 2023.

[13] F. Liu *et al.*, "Opt2skill: Imitating dynamically-feasible whole-body trajectories for versatile humanoid loco-manipulation," *IEEE Robotics and Automation Letters (RA-L)*, 2025.

[14] C. Zhang, W. Xiao, T. He, and G. Shi, "Wococo: Learning whole-body humanoid control with sequential contacts," in *Conference on Robot Rearning (CoRL)*, 2024.

[15] M. Liu *et al.*, "Visual whole-body control for legged loco-manipulation," in *Conference on Robot Rearning (CoRL)*, 2024.

[16] C. Schwarke, V. Klemm, M. Van der Boon, M. Bjelonic, and M. Hutter, "Curiosity-driven learning of joint locomotion and manipulation tasks," in *Conference on Robot Rearning (CoRL)*, 2023.

[17] J. Dao, H. Duan, and A. Fern, "Sim-to-real learning for humanoid box loco-manipulation," in *International Conference on Robotics and Automation (ICRA)*, 2024.

[18] Y. Zhang *et al.*, "Falcon: Learning force-adaptive humanoid loco-manipulation," *arXiv preprint arXiv:2505.06776*, 2025.

[19] C. Lu *et al.*, "Mobile-television: Predictive motion priors for humanoid whole-body control," in *International Conference on Robotics and Automation (ICRA)*, 2025.

[20] Q. Ben, F. Jia, J. Zeng, J. Dong, D. Lin, and J. Pang, "Homie: Humanoid loco-manipulation with isomorphic exoskeleton cockpit," in *Robotics: Science and Systems (RSS)*, 2025.

[21] Y. Xue, W. Dong, M. Liu, W. Zhang, and J. Pang, "A unified and general humanoid whole-body controller for versatile locomotion," in *Robotics: Science and Systems (RSS)*, 2025.

[22] J. Merel *et al.*, "Catch & carry: Reusable neural controllers for vision-guided whole-body tasks," *ACM Transactions on Graphics (TOG)*, 2020.

[23] M. Hassan, Y. Guo, T. Wang, M. Black, S. Fidler, and X. B. Peng, "Synthesizing physical character-scene interactions," in *ACM SIGGRAPH Conference*, 2023.

[24] Z. Xie, J. Tseng, S. Starke, M. van de Panne, and C. K. Liu, "Hierarchical planning and control for box loco-manipulation," *ACM Computer Graphics and Interactive Techniques*, 2023.

[25] J. Gao *et al.*, "Coohoi: Learning cooperative human-object interaction with manipulated object dynamics," *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

[26] Z. Xiao *et al.*, "Unified human-scene interaction via prompted chain-of-contacts," in *International Conference on Learning Representations (ICLR)*, 2024.

[27] L. Pan *et al.*, "Synthesizing physically plausible human motions in 3d scenes," in *International Conference on 3D Vision (3DV)*, 2024.

[28] L. Pan *et al.*, "Tokenhsi: Unified synthesis of physical human-scene interactions through task tokenization," in *Computer Vision and Pattern Recognition Conference (CVPR)*, 2025.

[29] N. Mahmood, N. Ghorbani, N. F. Troje, G. Pons-Moll, and M. J. Black, "Amass: Archive of motion capture as surface shapes," in *International Conference on Computer Vision (ICCV)*, 2019.

[30] M. Hassan *et al.*, "Stochastic scene-aware motion prediction," in *International Conference on Computer Vision (ICCV)*, 2021.

[31] X. B. Peng, Z. Ma, P. Abbeel, S. Levine, and A. Kanazawa, "Amp: Adversarial motion priors for stylized physics-based character control," *ACM Transactions on Graphics (ToG)*, 2021.

[32] A. Escontrela *et al.*, "Adversarial motion priors make good substitutes for complex reward functions," in *International Conference on Intelligent Robots and Systems (IROS)*, 2022.

[33] I. Mason, S. Starke, and T. Komura, "Real-time style modelling of human locomotion via feature-wise transformations and local motion phases," *ACM Computer Graphics and Interactive Techniques*, 2022.

[34] W. Wang *et al.*, "Sims: Simulating stylized human-scene interactions with retrieval-augmented script generation," in *International Conference on Computer Vision (ICCV)*, 2025.

[35] A. Allshire *et al.*, "Visual imitation enables contextual humanoid control," in *Conference on Robot Rearning (CoRL)*, 2025.

[36] H. Xue *et al.*, "Leverb: Humanoid whole-body control with latent vision-language instruction," *arXiv preprint arXiv:2506.13751*, 2025.

[37] X. B. Peng, P. Abbeel, S. Levine, and M. Van de Panne, "Deepmimic: Example-guided deep reinforcement learning of physics-based character skills," *ACM Transactions On Graphics (TOG)*, 2018.

[38] Y. Wang *et al.*, "Skillmimic: Learning basketball interaction skills from demonstrations," in *Computer Vision and Pattern Recognition Conference (CVPR)*, 2025.

[39] R. Yu *et al.*, "Skillmimic-v2: Learning robust and generalizable interaction skills from sparse and noisy demonstrations," in *ACM SIGGRAPH Conference*, 2025.

[40] Z. Luo *et al.*, "Universal humanoid motion representations for physics-based control," in *International Conference on Learning Representations (ICLR)*, 2023.

[41] L. Ma *et al.*, "Styleloco: Generative adversarial distillation for natural humanoid robot locomotion," *arXiv preprint arXiv:2503.15082*, 2025.

[42] S. Lin, G. Qiao, Y. Tai, A. Li, K. Jia, and G. Liu, "Hwc-loco: A hierarchical whole-body control approach to robust humanoid locomotion," *arXiv preprint arXiv:2503.00923*, 2025.

[43] J. Shi *et al.*, "Adversarial locomotion and motion imitation for humanoid policy learning," *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

[44] T. He *et al.*, "Omnih2o: Universal and dexterous human-to-humanoid whole-body teleoperation and learning," in *Conference on Robot Rearning (CoRL)*, 2024.

[45] Z. Zhang *et al.*, "Unleashing humanoid reaching potential via real-world-ready skill space," *arXiv preprint arXiv:2505.10918*, 2025.

[46] Z. Su *et al.*, "Hitter: A humanoid table tennis robot via hierarchical planning and learning," *arXiv preprint arXiv:2508.21043*, 2025.

[47] Z. Zhuang, S. Yao, and H. Zhao, "Humanoid parkour learning," in *Conference on Robot Rearning (CoRL)*, 2024.

[48] Y. Ma, A. Cramariuc, F. Farshidian, and M. Hutter, "Learning coordinated badminton skills for legged manipulators," *Science Robotics*, 2025.

[49] Z. Fu, Q. Zhao, Q. Wu, G. Wetzstein, and C. Finn, "Humanplus: Humanoid shadowing and imitation from humans," in *Conference on Robot Rearning (CoRL)*, 2020.

[50] T. Lin, K. Sachdev, L. Fan, J. Malik, and Y. Zhu, “Sim-to-real reinforcement learning for vision-based dexterous manipulation on humanoids,” in *Conference on Robot Rearning (CoRL)*, 2025.

[51] J. Bjorck *et al.*, “Gr00t n1: An open foundation model for generalist humanoid robots,” *arXiv preprint arXiv:2503.14734*, 2025.

[52] R.-Z. Qiu *et al.*, “Humanoid policy˜ human policy,” *arXiv preprint arXiv:2503.13441*, 2025.

[53] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, “Fast-lio2: Fast direct lidar-inertial odometry,” *IEEE Transactions on Robotics*, 2022.

[54] L. Wang, X. Zhang, C. Li, D. He, Y. Pan, and J. Yi, “Super-lio: A robust and efficient lidar-inertial odometry system with a compact mapping strategy,” *arXiv preprint arXiv:2509.05723*, 2025.

[55] H. Wang *et al.*, “Beamdojo: Learning agile humanoid locomotion on sparse footholds,” in *Robotics: Science and Systems (RSS)*, 2025.

[56] J. Ren *et al.*, “Vb-com: Learning vision-blind composite humanoid locomotion against deficient perception,” *arXiv preprint arXiv:2502.14814*, 2025.

[57] J. Long *et al.*, “Learning humanoid locomotion with perceptive internal model,” in *International Conference on Robotics and Automation (ICRA)*, 2025.

[58] Y. Li *et al.*, “Clone: Closed-loop whole-body humanoid teleoperation for long-horizon tasks,” *Conference on Robot Rearning (CoRL)*, 2025.

[59] Y. Ze *et al.*, “Twist: Teleoperated whole-body imitation system,” in *Conference on Robot Rearning (CoRL)*, 2025.

[60] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li, “On the continuity of rotation representations in neural networks,” in *Computer Vision and Pattern Recognition Conference (CVPR)*, 2019.

[61] L. Mescheder, A. Geiger, and S. Nowozin, “Which training methods for gans do actually converge?” In *International conference on machine learning (ICML)*, 2018.

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

[63] L. Pinto, M. Andrychowicz, P. Welinder, W. Zaremba, and P. Abbeel, “Asymmetric actor critic for image-based robot learning,” in *Robotics: Science and Systems (RSS)*, 2018.

[64] T. Kobayashi, “L2c2: Locally lipschitz continuous constraint towards stable and smooth reinforcement learning,” in *International Conference on Intelligent Robots and Systems (IROS)*, 2022.

[65] J. Wang and E. Olson, “AprilTag 2: Efficient and robust fiducial detection,” in *International Conference on Intelligent Robots and Systems (IROS)*, 2016.

[66] 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 *International Conference on Intelligent Robots and Systems (IROS)*, 2017.

[67] V. Makoviychuk *et al.*, “Isaac gym: High performance gpu-based physics simulation for robot learning,” *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[68] G. Comanici *et al.*, “Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities,” *arXiv preprint arXiv:2507.06261*, 2025.

[69] Z. Shen *et al.*, “World-grounded human motion recovery via gravity-view coordinates,” in *SIGGRAPH Asia Conference*, 2024.

Appendix

A. Tasks

In this section, we provide detailed definitions of each task, including the composition of the reference motion dataset M, the task-specific observation o_t^G, and the task reward r_t^G.

1) ***Carry Box:***

The humanoid starts from a random position and is tasked with approaching and moving a box from a randomly initialized 3D location to a target 3D location. In simulation, two thin platforms are used to support the box, since both the initial and target heights are randomly generated.

Reference Motion Dataset The motion dataset for *Carry Box* consists of two parts. The first part, *Loco*, includes 11 motion sequences from the AMASS [29] dataset, covering basic locomotion behaviors such as standing, walking, and turning on flat ground. The second part, *Carry*, includes 3 sequences from the AMASS dataset and 2 video-based motion sequences, which were retargeted to SMPL motions using GVHMR [69] and subsequently refined by manually correcting certain joints to ensure better physical feasibility. For convenience in RSI, this dataset is further divided into three subsets: *pickUp*, *carryWith*, and *putDown*.

Task Observations The task-specific observation $\mathbf{o}_t^G \in \mathbb{R}^{15}$ comprises the following properties of the target box:

- Box shape $\mathbf{b}_t \in \mathbb{R}^3$
- Box position $\mathbf{p}_{b_t}^{o_t} \in \mathbb{R}^3$
- Box rotation $\mathbf{R}_{b_t}^{o_t} \in \mathbb{R}^6$
- Goal location of the box $\mathbf{p}_{b_t}^{g_t} \in \mathbb{R}^3$

Task Rewards We implement the multi-stage task reward function similar to TokenHSI [28]. The first stage aims to encourage the robot to walk toward the initial box:

$$r_t^{loco} = \begin{cases} 1.5, & \left\|\mathbf{p}_{xy}^{o_t} - \mathbf{p}_{xy}^{b_t}\right\| < 0.7 \\ 1.0 \exp\left(-5.0 \left\|0.85 - \mathbf{d}_t^* \cdot \dot{\mathbf{p}}_{xy}^{b_t}\right\|^2\right) + \\ \quad 0.5 \exp\left(-0.75 \left\|\Delta\theta(\mathbf{d}_t^*, \mathbf{d}_{b_t})\right\|\right), & \text{otherwise} \end{cases} \tag{8}$$

where $\mathbf{p}_{xy}^{o_t}$ and $\mathbf{p}_{xy}^{b_t}$ denote the 2D positions of the object and the robot base in the world frame, respectively. $\mathbf{d}_t^*$ is a horizontal unit vector pointing from $\mathbf{p}_{xy}^{b_t}$ to $\mathbf{p}_{xy}^{o_t}$, $\mathbf{d}_{b_t}$ is the 2D horizontal unit vector of the robot base orientation in the world frame, and $a \cdot b$ represents the vector dot product. $\Delta\theta(\mathbf{d}_t^*, \mathbf{d}_{b_t})$ is the yaw error between target heading and root heading, defined as

$$\Delta\theta(\mathbf{a}, \mathbf{b}) = \arctan 2(\mathbf{a}_y, \mathbf{a}_x) - \arctan 2(\mathbf{b}_y, \mathbf{b}_x), \tag{9}$$

where $\mathbf{a}$ and $\mathbf{b}$ are 2D horizontal vectors. The second stage is to encourage the robot to pick up and move the box to its target location, which is defined:

$$r_t^{carry} = \begin{cases} 0.0, & \left\|\mathbf{p}_{xy}^{o_t} - \mathbf{p}_{xy}^{b_t}\right\| > 0.7 \\ 2.2, & \left\|\mathbf{p}_{xy}^{b_t} - \mathbf{p}_{xy}^{g_t}\right\| < 0.7 \\ 1.0 \exp\left(-5.0 \left\|0.85 - \mathbf{d}_t^{\#} \cdot \dot{\mathbf{p}}_{xy}^{b_t}\right\|^2\right) + \\ \quad 0.5 \exp\left(-0.75 \left\|\Delta\theta(\mathbf{d}_t^{\#}, \mathbf{d}_{b_t})\right\|\right) + \\ \quad 0.7 \exp\left(-3.0 \left\|\mathbf{p}^{o_t} - \mathbf{p}^{hand_t}\right\|^2\right), & \text{otherwise} \end{cases} \tag{10}$$

where $\mathbf{p}_{xy}^{g_t}$ denotes the 2D positions of the goal in the world frame, $\mathbf{d}_t^{\#}$ is a horizontal unit vector pointing from $\mathbf{p}_{xy}^{b_t}$ to $\mathbf{p}_{xy}^{g_t}$, and $\mathbf{p}^{hand_t}$ denotes the mean 3D coordinates of the robot's two hands. The third term of r_t^{carry} encourages the robot to pick up the box using its hands. To further reinforce this behavior, we additionally reward the lifting height during the pickup stage, defined as:

$$r_t^{pick} = \begin{cases} 0.0, & \left\|\mathbf{p}_{xy}^{o_t} - \mathbf{p}_{xy}^{b_t}\right\| > 0.7 \\ 2.0, & \left\|\mathbf{p}_{xy}^{b_t} - \mathbf{p}_{xy}^{g_t}\right\| < 0.7 \text{ or } \mathbf{p}_z^{o_t} > 0.75 \\ 2.0 \exp\left(-3.0 \left\|0.75 - \mathbf{p}_z^{o_t}\right\|\right), & \text{otherwise} \end{cases} \tag{11}$$

where $\mathbf{p}_z^{o_t}$ denotes the height of the box in the world frame. Additionally, we further design a reward function r_t^{put} to encourage the robot to accurately place the box at the target location:

$$r_t^{put} = \begin{cases} 0.0, & \left\|\mathbf{p}_{xy}^{b_t} - \mathbf{p}_{xy}^{g_t}\right\| > 0.7 \\ 2.0, & \left\|\mathbf{p}^{o_t} - \mathbf{p}^{g_t}\right\| < 0.05 \\ 1.0 \exp\left(-10.0 \left\|\mathbf{p}^{o_t} - \mathbf{p}^{g_t}\right\|\right) + \\ \quad 1.0 \exp\left(-3.0 \left(\mathbf{p}_z^{o_t} - \mathbf{p}_z^{g_t}\right)\right), & \text{otherwise} \end{cases} \tag{12}$$

where $\mathbf{p}_z^{g_t}$ denotes the height of the goal in the world frame. Therefore, the total task reward function for *Carry Box* be formulated as:

$$r_t^{G\_carryBox} = r_t^{loco} + r_t^{carry} + r_t^{pick} + r_t^{put}. \tag{13}$$

Scene Randomization We randomize the task scene along the following four dimensions:

- The 2D position of the box and the target relative to the robot's initial position is uniformly sampled from $[-4.0, 4.0]\,\mathrm{m}$ relative to the robot's initial base position.
- The height position of both the box and the target is uniformly sampled from $[0.0, 0.6]\,\mathrm{m}$ above the ground.
- The box size is randomized, with the width uniformly sampled from $[0.2, 0.5]\,\mathrm{m}$ and the height uniformly sampled from $[0.15, 0.35]\,\mathrm{m}$.
- The box density is uniformly sampled from $[10, 100]\,\mathrm{kg/m^3}$.

Observation Mask Strategy As discussed in Sec. IV-B, we align the simulation training with the real-world deployment by masking object observations $\mathbf{p}_{b_t}^{o_t}$ and $\mathbf{R}_{b_t}^{o_t}$ when the box is *out of view* during the grasping stage of dynamic objects. We define an object as *out of view* in simulation if it satisfies one of the following conditions:

- Facing condition: The surface normal of the box must face toward the camera, i.e., the angle between the viewing direction and the surface normal is within $(60° + \Delta)$, where the offset Δ is uniformly sampled from $[-10°, 10°]$.
- FOV condition: All tag positions of the box must lie within the camera's field-of-view (FOV), constrained by both the horizontal and vertical FOV angles.
- Distance condition: The mean tag position must lie within $2.5\,\mathrm{m}$ range from the camera.

*2) **Sit Down and Lie Down**:* The humanoid starts from a random position and is tasked with approaching a fixed chair or bed surface to perform a sitting or lying action. In the simulation, a thin platform and a box are used to support sitting and lying behaviors, respectively.

Reference Motion Dataset For locomotion, we use the same *Loco* dataset as in the *Carry Box* task. For the *Sit Down* task, we additionally select four sitting sequences from SAMP [30], and for the *Lie Down* task, we additionally select six lying sequences from SAMP.

Task Observations The task-specific observation $\mathbf{o}_t^G \in \mathbb{R}^9$ comprises the following properties of the target chair/bed:

- Chair/bed position $\mathbf{p}_{b_t}^{o_t} \in \mathbb{R}^3$
- Chair/bed rotation $\mathbf{R}_{b_t}^{o_t} \in \mathbb{R}^6$

Task Rewards For the locomotion stage, both tasks share the same locomotion reward r_t^{loco} as in the *Carry Box* task. The second stage encourages the robot to sit down on the chair or bed, with the sitting reward defined as:

$$r_t^{sit} = \begin{cases} 0.0, \ \left\| \mathbf{p}_{xy}^{o_t} - \mathbf{p}_{xy}^{b_t} \right\| > 0.7 \\ 1.0 \exp\left(-3.0 \left\| \mathbf{p}^{o_t} - \mathbf{p}^{b_t} \right\| \right) + \\ \quad 1.0 \exp\left(-5.0 \left(\mathbf{p}_z^{o_t} - \mathbf{p}_z^{b_t} \right) \right) + \\ \quad 1.0 \exp\left(-0.75 \left\| \Delta\theta(\mathbf{d}_{o_t}, \mathbf{d}_{b_t}) \right\| \right), \ \text{otherwise} \end{cases} \tag{14}$$

where $\mathbf{p}^{o_t}$ and $\mathbf{p}^{b_t}$ represent the 3D positions of the object (chair/bed) surface center and the robot base in the world frame, respectively, while $\mathbf{d}_{o_t}$ and $\mathbf{d}_{b_t}$ denote the 2D horizontal unit vectors of the object and the robot base orientations in the world frame. The total reward function for *Sit Down* is then given by:

$$r_t^{G\_sitDown} = r_t^{loco} + r_t^{sit}. \tag{15}$$

For *Lie Down*, once the robot has successfully sat on the bed, we introduce an additional reward to encourage lying down while facing upward toward the sky, combined with r_t^{sit} and defined as:

$$r_t^{lie} = \begin{cases} r_t^{sit}, \ \left\| \mathbf{p}_{xy}^{o_t} - \mathbf{p}_{xy}^{b_t} \right\| < 0.3 \text{ and } \left(\mathbf{p}_z^{o_t} - \mathbf{p}_z^{b_t} \right) < 0.05 \\ 3.0 + 0.5 \exp\left(-0.75 \left\| \mathbf{D}_{world\_z} \cdot \mathbf{D}_{b_t}) \right\| \right) + \\ \quad 0.5 \exp\left(-2 \left\| \Delta\theta(\mathbf{d}_{o_t}^{\perp}, \mathbf{d}_t^{\dagger}) \right\| \right), \ \text{otherwise} \end{cases} \tag{16}$$

Here, $\mathbf{D}_{world\_z} \in \mathbb{R}^3$ denotes the global vertical direction $[0, 0, 1]$, and $\mathbf{D}_{b_t} \in \mathbb{R}^3$ denotes the robot's upward direction vector. The first term encourages the robot to face upward toward the sky. $\mathbf{d}_{o_t}^{\perp}$ is the 2D unit vector perpendicular to $\mathbf{d}_{o_t}$, and $\mathbf{d}_t^{\dagger}$ is the horizontal unit vector pointing from the head to the robot base. The second term encourages the robot to align parallel with the bed edge. Then the total reward function for *Sit Down* is then given by:

$$r_t^{G\_lieDown} = r_t^{loco} + r_t^{lie}. \tag{17}$$

Scene Randomization We randomize the task scene along the following three dimensions:

- The 2D position of the chair or bed is uniformly sampled within $[-5.0, 5.0]\,\mathrm{m}$ relative to the robot's initial base position.
- The height of both the chair and the bed is uniformly sampled from $[0.2, 0.5]\,\mathrm{m}$ above the ground.
- The size of the chair and bed is randomized: the length and width of the chair are uniformly sampled from $[0.3, 0.6]\,\mathrm{m}$, while the bed length is uniformly sampled from $[1.2, 3.2]\,\mathrm{m}$ and its width from $[0.38, 0.63]\,\mathrm{m}$.

*3) **Stand Up**:* The humanoid starts in a seated position on the fixed chair and is tasked with standing up and walking toward a designated target location. Similarly, a thin platform is used to support stable sitting behaviors in simulation.

Reference Motion Dataset For locomotion, we use the same *Loco* dataset as in the *Carry Box* task. For the *Stand Up* task, we additionally select two getting up sequences from SAMP.

In addition, to ensure initialization stability, we pre-collected a set of stable sitting poses on chairs of different heights using the sitting policy. These poses were used for initialization during training.

Task Observations The task-specific observation $\mathbf{o}_t^G \in \mathbb{R}^{12}$ comprises the following components:

- Chair position $\mathbf{p}_{b_t}^{o_t} \in \mathbb{R}^3$
- Chair rotation $\mathbf{R}_{b_t}^{o_t} \in \mathbb{R}^6$
- Target position $\mathbf{p}_{b_t}^{g_t} \in \mathbb{R}^3$

Task Rewards In the first stage, the robot is encouraged to stand up to a target height, defined as:

$$r_t^{standup} = \begin{cases} 3.0, \ \mathbf{p}_z^{b_t} > 0.72 \\ 3.0 \exp\left(-5.0 \left(0.72 - \mathbf{p}_z^{b_t} \right) \right). \ \text{otherwise} \end{cases} \tag{18}$$

Once the robot has reached the target height, it is encouraged to walk toward the goal position, defined as:

$$\begin{aligned} r_t^{loco\_tar} = {} & 0.5 \exp\left(-5.0 \left\| 0.85 - \mathbf{d}_t' \cdot \dot{\mathbf{p}}_{xy}^{b_t} \right\|^2 \right) + \\ & 0.5 \exp\left(-0.75 \left\| \Delta\theta(\mathbf{d}_t', \mathbf{d}_{b_t}) \right\| \right), \end{aligned} \tag{19}$$

where $\mathbf{d}_t'$ is a horizontal unit vector pointing from $\mathbf{p}_{xy}^{b_t}$ to $\mathbf{p}_{xy}^{g_t}$. The total reward function for *Stand Up* is then given by:

$$r_t^{G\_standUp} = r_t^{standup} + r_t^{loco\_tar}. \tag{20}$$

Scene Randomization We randomize the task scene along the following three dimensions:

- The 2D position of the target is uniformly sampled within $[-5.0, 5.0]\,\mathrm{m}$ relative to the robot's initial base position.
- The height of the chair is uniformly sampled from $[0.2, 0.6]\,\mathrm{m}$ above the ground.
- The length and width of the chair is uniformly sampled from $[0.38, 0.63]\,\mathrm{m}$.

Configuration In the original setting, we use a standing pose as the default pose. When initializing the robot from a seated pose, the policy often causes an abrupt upward jerk to transition into standing, disrupting training and causing instability. Therefore, for the *Stand Up*, we set the default pose to a predefined seated pose, detailed in Table V, to ensure a stable initial state.

TABLE V: Default Pose Configuration for the *Stand Up* Task

Term	Value	Term	Value
left hip pitch joint	-1.2	right hip pitch joint	-1.2
left hip roll joint	0.2	right hip yaw joint	-0.2
left hip yaw joint	0.0	right hip yaw joint	0.0
left knee joint	1.2	right knee joint	1.2
left ankle pitch joint	0.0	right ankle pitch joint	0.0
left ankle roll joint	0.0	right ankle roll joint	0.0
left shoulder pitch joint	0.2	right shoulder pitch joint	0.2
left shoulder roll joint	0.8	right shoulder roll joint	-0.8
left shoulder yaw joint	-0.7	right shoulder yaw joint	0.7
left elbow joint	-0.3	right elbow joint	-0.3
left wrist roll joint	0.0	right wrist roll joint	0.0
left wrist pitch joint	0.0	right wrist pitch joint	0.0
left wrist yaw joint	0.0	right wrist yaw joint	0.0
waist yaw joint	0.0	waist roll joint	0.0
waist pitch joint	0.6		

*4) **Stylized Locomotion**:* The humanoid is tasked with tracking the given command $\mathbf{c}_t = [\mathbf{v}_x^c, \mathbf{v}_y^c, \boldsymbol{\omega}_{\text{yaw}}^c] \in \mathbb{R}^3$ (denote the linear velocities in the longitudinal and lateral directions, and the angular velocity in the horizontal plane, respectively) while walking with a stylized gait, such as dinosaur-like walking or high-knee stepping.

Reference Motion Dataset We select two motion styles, *Dinosaur* and *HighKnees*, from the 100STYLE dataset [33]. Each style includes three sequences: forward walking, backward walking, and sidestep walking.

Task Rewards The only task reward for *Stylized Locomotion* is to track the given linear and angular velocities, defined as:

$$r_t^{G\_styleLoco} = 1.0 \exp\left(-4 \left\|\mathbf{v}_{xy} - \mathbf{v}_{xy}^c\right\|^2\right) + 0.5 \exp\left(-4 \left(\boldsymbol{\omega}_{\text{yaw}} - \boldsymbol{\omega}_{\text{yaw}}^c\right)^2\right). \tag{21}$$

B. Training Details

1) Regularization Rewards: The regularization reward r_t^R is summarized in Table VI.

TABLE VI: Regularization Reward Functions

Term	Weight	Term	Weight
dof velocity	$-2e-4$	torques	$-1e-4$
dof acceleration	$-1e-7$	torque limits	-0.1
dof position limits	-5.0	action rate	-0.03
dof velocity limits	$-1e-3$		

2) Domain Randomization: To enhance robustness and facilitate sim-to-real transfer, we employ domain randomization, summarized in Table VII.

3) Hyperparameters: The hyperparameters used for training is summarized in Table VIII.

C. Deployment Details

To support the Intel RealSense D455 camera mounted on the humanoid's head, we designed a 3D-printed camera bracket. The bracket is fixed to the torso link with an offset of $(0.08, 0.01, 0.40)$ m and rotated by approximately 40° about the pitch axis, as shown in Fig. 6.

TABLE VII: Domain Randomization Settings

Term	Value
Observations	
angular velocity noise	$\mathcal{U}(-0.3, 0.3)$ rad/s
joint position noise	$\mathcal{U}(-0.02, 0.02)$ rad/s
joint velocity noise	$\mathcal{U}(-2.0, 2.0)$ rad/s
projected gravity noise	$\mathcal{U}(-0.05, 0.05)$ rad/s
FK noise	$\mathcal{U}(-0.05, 0.05)$ m
Humanoid Physical Properties	
actuator offset	$\mathcal{U}(-0.05, 0.05)$ rad
motor strength noise	$\mathcal{U}(0.9, 1.1)$
payload mass	$\mathcal{U}(-2.0, 2.0)$ kg
center of mass displacement	$\mathcal{U}(-0.05, 0.05)$ m
Kp, Kd noise factor	$\mathcal{U}(0.85, 1.15)$
Object Dynamics	
box friction factor	$\mathcal{U}(0.5, 1.2)$
box restitution factor	$\mathcal{U}(0.0, 0.2)$
platform friction factor	$\mathcal{U}(0.5, 1.2)$
Object Localization	
position offset	$\mathcal{U}(-0.05, 0.05)$ m
position noise	$\mathcal{U}(-0.05, 0.05)$ m
rotation offset	$\mathcal{U}(-5.0, 5.0)^\circ$
rotation noise	$\mathcal{U}(-5.0, 5.0)^\circ$

TABLE VIII: Hyperparameters

Hyperparameter	Value
General	
num of robots	4096
num of steps per iteration	100
num of epochs	5
gradient clipping	1.0
adam epsilon	$1e-8$
PPO	
clip range	0.2
entropy coefficient	0.01
discount factor γ	0.99
GAE balancing factor λ	0.95
desired KL-divergence	0.01
actor and double critic NN	MLP, hidden units [512, 256, 256]
PhysHSI	
reward coefficient w^S (general)	0.3
reward coefficient w^G, w^R	0.7, 0.7
gradient penalty w^{gp}	1.0
distance threshold ϵ	0.6
AMP discriminator NN	MLP, hidden units [512, 256, 256]

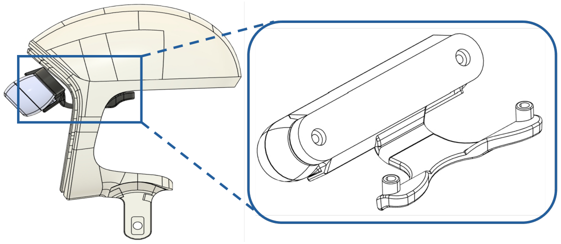

Fig. 6: 3D model of the D455 camera bracket.

D. Evaluation Details

1) Baseline Implementation Details: For **RL-Rewards**, we replace the style reward r_t^S with explored RL-based gait rewards [55, 57] to regularize the gait during task execution. The total reward function is formulated as

$$r_t^{RL} = r_t^{Gait} + r_t^R + r_t^G, \tag{22}$$

where r_t^R and r_t^G are the same as in PhysHSI. The gait reward terms are summarized in Table IX.

For **Tracking-Based**, we employ the tracking reward

TABLE IX: Gait Reward Functions

Term	**Weight**	**Term**	**Weight**
base height	-10.0	feet clearance	-0.5
z velocity	-2.0	feet air time	0.05
roll-pitch velocity	-0.05	feet distance	0.5
orientation	-1.0	hip joint deviation	-0.5

r_t^{Track}, primarily adapted from [5], together with the regularization reward r_t^R. The tracking reward terms are summarized in Table X.

TABLE X: Tracking Reward Functions

Term	**Weight**	**Term**	**Weight**
body position	1.0	base linear velocity	0.5
body rotation	0.5	end-effector position	1.5
base position	1.0	joint position	1.0
base rotation	0.5	object position (*Carry Box*)	1.0

2) Success Rate Evaluation: The success criteria for each task are defined as follows:

- *Carry Box*: The box is placed at the target position with a distance error of less than $0.1\,\mathrm{m}$.
- *Sit Down* and *Lie Down*: The robot's base position is within $0.1\,\mathrm{m}$ of the target location, with its heading aligned to the chair/bed orientation. For *Lie Down*, the body direction from head to feet must also be parallel to the bed, both with a tolerance of $15°$.
- *Stand Up*: The robot successfully stands up from the chair with a base height exceeding $0.72\,\mathrm{m}$, and then reaches the target position with a distance error of less than $0.3\,\mathrm{m}$.

3) Human-Likeness Score Evaluation: We evaluate the human-likeness score S_{human} using Gemini-2.5-Pro [68]. The model is prompted with task descriptions and experimental trajectories, and outputs a human-likeness score in the range $[0, 5]$ for each demonstration clip. Below is an example prompt used for evaluating the *Lie Down* task:

```
I will provide you with three videos of a robot
    performing a lie-down task, named
    LieDown_PhysHSI_1, LieDown_Mimic_1, and
    LieDown_RL_1.

Please analyze and compare them based on the
    criterion of Naturalness. For this task,
    Naturalness is defined by how closely the robot
    's movement resembles a natural, human-like
    action. When evaluating, please consider these
    specific aspects:

1. Fluidity and Smoothness: Is the motion
    continuous, or is it jerky and segmented?
2. Stability and Balance: Does the robot appear
    stable and in control, or does it look wobbly
    and at risk of falling?
3. Plausibility of Strategy: Does the robot use its
     limbs and body in a way a human would (e.g.,
    using hands for support, bending knees,
    controlled descent)?

Please provide a Naturalness score out of 5 (
    decimals are allowed) for each video in a table
    . After the scores, write a summary explaining
    the key differences and justifying your ratings
     based on the aspects mentioned above.
```