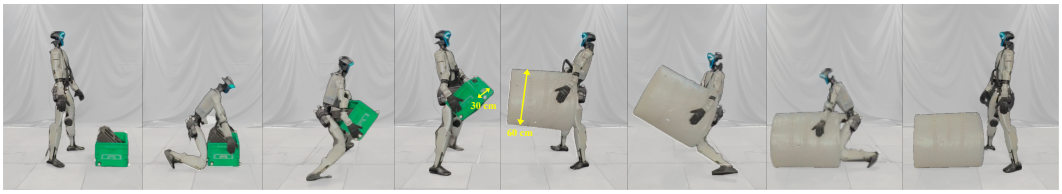
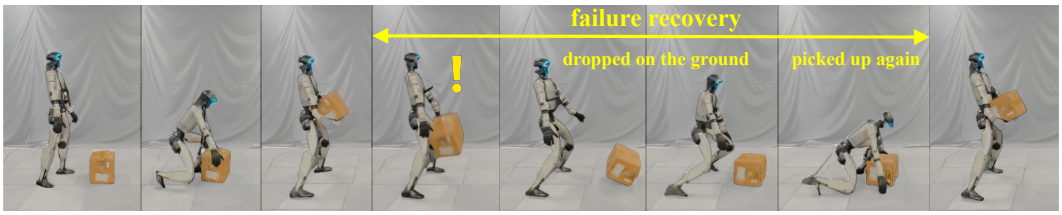
LESSMIMIC: Versatile Humanoid–Object Interaction with Unified Distance Field Representations

Yutang Lin[*,1,2,3,5,7,8] **Jieming Cui**[*,1,2,3,5,7,8] **Yixuan Li**[4,2,5]
Wei Liang[4] **Baoxiong Jia**[†,2,5] **Yixin Zhu**[†,3,1,5,6,7,8] **Siyuan Huang**[†,2,5]

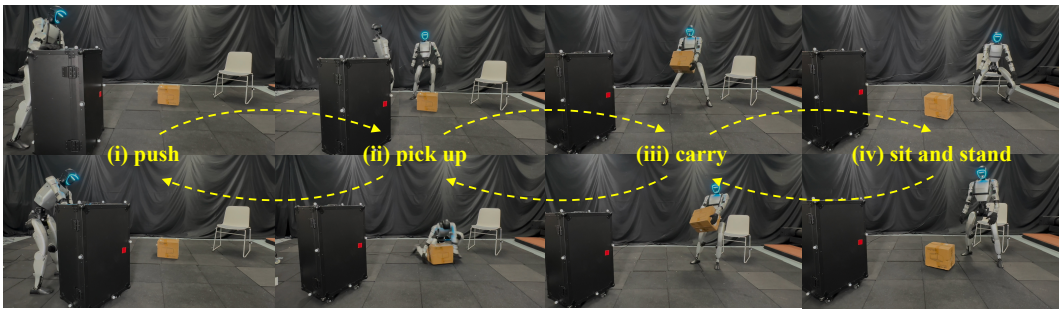
[1]Institute for AI, Peking University [2]Beijing Institute for General Artificial Intelligence (BIGAI)
[3]School of Psychological and Cognitive Sciences, Peking University
[4]School of Computer Science and Technology, Beijing Institute of Technology
[5]State Key Lab of General AI [6]XS Vision
[7]Beijing Key Laboratory of Behavior and Mental Health, Peking University
[8]Embodied Intelligence Lab, PKU-Wuhan Institute for Artificial Intelligence
[*]Equal contribution [†]Corresponding author



(a) **Shape and size generalization.** One policy lifts objects ranging from a 30 cm box to a 60 cm cylinder.



(b) **Failure recovery.** After a perturbation, the humanoid restarts the pickup from the new object location.



(c) **Skill composition.** One policy executes Push, PickUp, Carry, and SitStand in sequence.

Figure 1: **Diverse humanoid–object interaction with one LESSMIMIC policy.** The DF-conditioned policy (a) generalizes to unseen object shapes and scales, (b) recovers from failures through continuous geometric feedback, and (c) composes heterogeneous interaction skills.

Abstract

Versatile whole-body interaction between humanoid robots and everyday objects is a central goal of embodied intelligence. Existing methods often rely on motion-reference inputs or task-specific rewards, which couple each policy to particular motion scripts, object geometries, and contact timings. This coupling limits both geometric generalization and flexible interaction control, in which simple commands

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

specify the motion intent and the object geometry determines how contact is executed. Here we present **LESSMIMIC**, a framework for versatile humanoid–object interaction that requires no motion reference at inference. A single whole-body policy is steered by root-velocity commands and an interaction-type flag, and it is conditioned on a compact interaction representation built from short histories of surface distances and local surface directions queried from a Distance Field (DF). This geometry-conditioned interface lets the commands set the motion intent while contact behavior adapts to the local object geometry. Visual distillation further transfers the interface to egocentric depth, which yields a variant that deploys without Motion Capture (MoCap). Across object scales from $0.4\times$ to $1.6\times$, **LESSMIMIC** performs robustly on four interaction tasks (PickUp, SitStand, Push, and Carry), whereas motion-conditioned baselines degrade sharply away from the training scale. The same policy also composes heterogeneous skills in sequence, succeeding on 62.1% of five-task sequences and on 23.5% of sequences with 15 task instances. Grounding interaction control in local geometry rather than in motion references thus offers a path toward humanoid–object control that generalizes across geometries, follows simple commands, and composes heterogeneous skills.

1 Introduction

Consider a humanoid tidying a room. It pushes a chair aside, picks up a box, carries the box across the room, and sits down to rest. Each step involves different contact patterns, object geometries, and body configurations, yet a person can perform the whole sequence without following a fixed motion script. Recent humanoid controllers achieve impressive locomotion and dynamic motion [5, 10, 23, 24, 27, 61], but reproducing this versatility with varied objects on hardware remains an open problem in embodied intelligence [4, 19, 33].

The core obstacle is representational. Current approaches differ mainly in whether the deployed policy requires motion references, and both choices impose limitations. Motion-conditioned methods [27, 50, 54, 61, 62] condition the policy on recorded demonstrations and achieve high-fidelity motion. This formulation, however, entangles object geometry, contact timing, and body motion with specific reference trajectories. The policy can then become a geometric specialist that loses flexible interaction control, because every deviation from the reference is penalized as a tracking error, even when the deviation is needed to adapt to a new object, a changed root command, or accumulated state drift. Recent variants add higher-level planners or human-in-the-loop teleoperation [6, 23, 31, 53, 56, 58], but these mainly compensate for the same representational bottleneck, at the cost of additional perceptual complexity or human involvement. Motion-free-at-inference methods remove motion references from the deployed policy and can be steered more flexibly, but without a principled interaction signal they often rely on task-specific rewards or handcrafted observations [8, 47], which yields isolated policies that do not naturally compose across heterogeneous skills. What is missing is a geometry-aware representation, shared across tasks, that decouples interaction logic from specific motion patterns, so that simple commands specify the motion intent and the local geometry determines how contact is executed.

We argue that the Distance Field (DF) [34] provides a natural basis for this representation. A DF maps each query point to its distance from the nearest object surface and thereby gives compact proximity information and, where the field is differentiable, local surface-direction cues. Near non-smooth regions, numerical direction estimates still provide practical contact-relevant cues, so the representation does not require global differentiability. Compared with point clouds, voxels, or implicit neural geometry, DFs offer efficient, geometry-aligned queries suited to high-frequency whole-body control. Expressed as local distances and surface directions, and combined with decomposed link velocities during post-training, DFs supply the interaction signal for geometry-aware contact adaptation while reducing the dependence on absolute object coordinates and demonstration-specific trajectories.

We introduce **LESSMIMIC**, a framework for versatile humanoid–object interaction that is motion-free at inference, meaning that the deployed policy consumes no reference joint poses, future body motion, or contact trajectories. A root-velocity command and an interaction-type flag steer the policy, while local DF observations ground its contact behavior in the current object geometry. **LESSMIMIC** encodes short histories of DF-derived surface distances and directions into a compact interaction representation and trains a single whole-body policy in three stages. Teacher-guided behavior cloning

gives a stable, contact-rich initialization, Reinforcement Learning (RL) under randomized object geometries uses Adversarial Interaction Priors (AIP) as a cross-task interaction reward, and visual distillation approximates the geometric interface from egocentric depth.

The result is one low-level policy interface for command-steered humanoid–object control (Fig. 1). **LESSMIMIC** uses this interface unchanged across four tasks (PickUp, SitStand, Push, and Carry) and object scales from $0.4\times$ to $1.6\times$; on PickUp at $0.4\times$, it succeeds in 63.0% of episodes, against 0.0% for our reproductions of the motion-conditioned HDMI [50] and ResMimic [62]. In sequential multi-skill evaluation, the same policy executes externally commanded interaction modes without skill-specific policies and succeeds on 62.1% of five-task sequences and on 23.5% of sequences with 15 task instances, whereas most ablated variants collapse.

Our contributions are threefold.

- **A unified DF-based interaction representation.** Its geometry-aware robot–object cues let one policy adapt its behavior to object scale, shape, and surface geometry.
- **A learning pipeline that is motion-free at inference.** It transfers physically valid contact behavior from a motion-imitation teacher into a command-steered policy and improves generalization under randomized object geometries with AIP-guided RL.
- **A visual distillation scheme.** It transfers the DF-based interface from privileged object geometry to egocentric depth, which yields a variant that deploys without Motion Capture (MoCap).

2 Related Work

2.1 Motion-Conditioned Humanoid–Object Interaction

RL has enabled increasingly capable whole-body humanoid control for locomotion, dynamic motion, and physical interaction [7, 10, 15, 32, 43, 48, 51]. A dominant paradigm conditions humanoid–object policies on reference motions or target poses to stabilize learning and produce plausible behavior [1, 27, 50, 53, 56, 61, 62]. Such motion-centric formulations tie contact timing, object geometry, and motion topology to the reference data, which limits generalization when object shape, scale, or configuration changes.

Table 1: **Comparison of humanoid–object interaction methods.** Check marks indicate inference without motion or pose references, a unified geometric observation, and multi-skill learning in one policy.

Method	No motion/pose ref. input	Unified geometric observation	One-policy multi-skill learning
HDMI [50]	✗	✗	✓
ResMimic [62]	✗	✗	✓
CLONE [23]	✗	✗	✓
HumanX [49]	✓	✗	✗
HAIC [21]	✗	✓	✗
PhysHSI [47]	✓	✗	✗
Ours	✓	✓	✓

Other systems broaden skill acquisition through scalable data generation, teleoperation, or human-in-the-loop correction [7, 10, 11, 18, 22, 23, 31, 49, 57, 58]. They increase the availability and diversity of motion supervision but provide no unified geometric policy interface that serves heterogeneous interaction skills without motion or pose references.

Table 1 summarizes these distinctions. HDMI and ResMimic learn multiple skills within one policy, but their deployed interface remains reference-conditioned, so composing a new heterogeneous sequence requires corresponding motion and object references, including the transitions between skills. CLONE supports sequential interaction through teleoperation. HumanX learns a broad range of humanoid skills from human videos, but it targets scalable skill acquisition rather than a steerable low-level interaction interface. HAIC [21] uses geometric observations yet remains motion-conditioned at inference, and PhysHSI is motion-free at inference but uses no unified geometric observation. **LESSMIMIC** is motion-free at inference, uses a unified DF-based geometric observation, and switches between heterogeneous skills online from the interaction-type flag and root commands, so no full-body reference has to be collected for each composed sequence.

2.2 Motion-Free-at-Inference Humanoid–Object Interaction

Removing motion or pose references at inference is important for flexible humanoid interaction control [30]. Classical model-based control and online optimization generate behavior from objectives without prerecorded motion references [12, 13, 37, 40, 41], but they often require accurate models and carefully designed objectives. Learning-based methods instead condition policies on task-

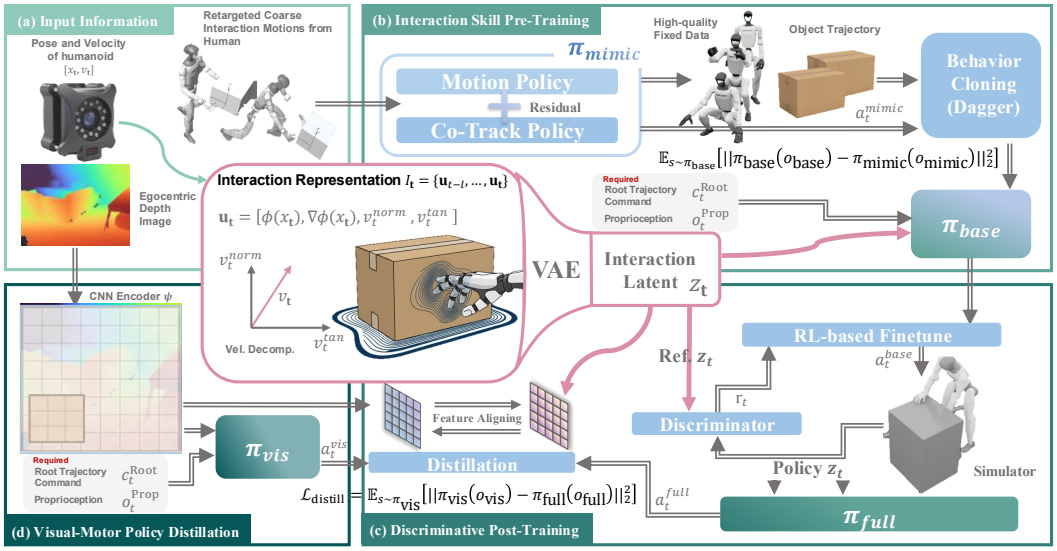


Figure 2: **Training pipeline of LESSMIMIC.** (a) DF-based robot–object cues from MoCap or egocentric depth are encoded into an interaction latent z_t. (b) A motion-tracking teacher π_{mimic} provides physically feasible demonstrations for behavior cloning of π_{base}, which uses the inference-time observations. (c) π_{base} is fine-tuned with AIP-guided RL under randomized geometries into π_{full}. (d) π_{full} is distilled into a vision policy π_{vis} for deployment without MoCap.

relevant observations [8, 20, 24, 26, 36, 45, 47, 60]; PhysHSI [47], for example, studies motion-free humanoid–scene interaction. These methods reduce the reliance on deployed motion references but often remain tied to task-specific observations, rewards, dynamics assumptions, or specialized domains. **LESSMIMIC** uses one unified geometric observation across heterogeneous humanoid–object interaction skills.

2.3 Geometric Representations for Interaction

Interaction methods encode geometry as voxels and occupancy maps [14, 17, 29, 46, 52], points or meshes [2, 9, 16, 28, 35, 42, 59, 64], or implicit neural representations [3, 38, 44, 55]. These representations trade off geometric detail, query efficiency, and suitability for high-frequency contact-aware control. DFs have appeared in robotics, for example to represent articulated robot geometry [25], and a humanoid whole-body policy for embracing bulky objects [63] learns from object-level cues such as the distance to the object center. Prior work, however, has not used object-surface distance and direction queries at humanoid body links as a deployed policy observation for versatile humanoid–object interaction. **LESSMIMIC** queries the DF at a fixed set of humanoid links, so each query relates one link to the object surface rather than to the object as a whole. These queries form one unified interaction representation across heterogeneous skills, and AIP further regularizes the resulting interaction statistics.

3 The LESSMIMIC Framework

LESSMIMIC couples a DF-based interaction representation with a whole-body controller that is motion-free at inference (Fig. 2). This coupling supports scale generalization and the sequential composition of heterogeneous interaction skills. Section 3.1 defines the representation, and the three training stages follow in order, from interaction skill pre-training (Sec. 3.2) through discriminative post-training (Sec. 3.3) to visual-motor distillation (Sec. 3.4).

3.1 DF-Based Interaction Representation

LESSMIMIC describes interaction through local robot–object geometry rather than absolute motion trajectories. Let $\Phi : \mathbb{R}^3 \to \mathbb{R}_{\geq 0}$ be the unsigned Euclidean distance to the object surface. The DF is queried at a fixed set $\mathcal{K}$ of humanoid links consisting of both wrists, both ankles, and the pelvis. At time t, the query at the position $\mathbf{x}_{t,k}$ of each link $k \in \mathcal{K}$ returns the distance $d_{t,k} = \Phi(\mathbf{x}_{t,k})$ and a

local surface-direction estimate $\mathbf{n}_{t,k}$. Where Φ is differentiable, $\mathbf{n}_{t,k} = \nabla\Phi(\mathbf{x}_{t,k})/\|\nabla\Phi(\mathbf{x}_{t,k})\|_2$; near non-smooth regions, such as edges or points with non-unique projections, a normalized numerical estimate replaces the gradient. The method therefore relies on stable local direction cues rather than on global differentiability of the DF. A batched pass over these queries takes less than 9 ms on one RTX 5090, even for 1024 parallel environments with three objects (Tab. A1).

Each link contributes the feature

$$\mathbf{u}_{t,k} = [d_{t,k}, \mathbf{n}_{t,k}], \tag{1}$$

and the interaction history over a window of $l = 5$ steps is

$$I_t = \{\mathbf{u}_{\tau,k}\}_{\tau=t-l+1:t,\, k\in\mathcal{K}}. \tag{2}$$

This history is not a complete reconstruction of the object geometry but a summary of the contact-relevant local proximity and surface-direction cues for whole-body control. Because it is defined by local robot–object relations rather than by absolute object coordinates or reference trajectories, it reduces the dependence on object-specific demonstrations.

A Variational Auto-Encoder (VAE) encodes I_t into a compact 64-dimensional latent z_t, which gives the policy a fixed-size interaction representation and smooths noisy or discontinuous local DF queries (Sec. A.1). During discriminative post-training, AIP augments this latent with link velocities expressed in the local surface frame (Sec. 3.3).

3.2 Interaction Skill Pre-Training

Learning contact-rich whole-body interaction directly from sparse objectives is unstable, because meaningful rewards often appear only after balance, locomotion, and object contact are already coordinated. We therefore use a physics-based mimic teacher, but only as training supervision. Following ResMimic [62], the teacher π_{mimic} is a single multi-task policy that tracks multiple retargeted human–object interaction motions associated with three training object instances and the four task categories PickUp, SitStand, Push, and Carry. It receives the current reference-motion state together with privileged interaction observations such as body velocity, object pose, and contact, and produces feasible contact behavior under full simulation physics.

The student policy π_{base} is trained on the observation used at inference,

$$o_{\text{base}} = [o_{\text{prop}}, c_t^{\text{root}}, g_t, z_t], \tag{3}$$

where o_{prop} is proprioception, $c_t^{\text{root}} \in \mathbb{R}^2$ is the commanded planar root velocity, g_t is a one-hot interaction-type flag, and z_t is the DF-based interaction latent. The flag specifies the intended skill mode but carries no reference joint poses, future body motion, or contact trajectories. All three stages use the same Transformer-based whole-body controller architecture, whose input tokens encode these four components and whose output is the whole-body action (Sec. A.2).

We train π_{base} with DAgger-style behavior cloning [39], in which the student acts in simulation and the teacher provides corrective actions at the visited states. The objective is

$$\mathcal{L}_{\text{BC}} = \mathbb{E}_{s\sim\pi_{\text{base}}}\left[\|\pi_{\text{base}}(o_{\text{base}}) - \pi_{\text{mimic}}(o_{\text{mimic}})\|_2^2\right], \tag{4}$$

where o_{mimic} contains privileged reference-motion information unavailable to the deployed policy. Teacher and student run in parallel simulation environments, each pairing a training object with its reference motion or motions, and rollouts from all tasks and objects are sampled jointly to update the single π_{base}. This stage applies no object-scale randomization; geometry randomization enters only in discriminative post-training. Pre-training thus transfers contact-rich imitation supervision into a policy that is motion-free at inference.

3.3 Discriminative Post-Training

Pre-training provides physically feasible contact behavior, but the policy can still inherit object-specific biases from the teacher trajectories. We therefore fine-tune π_{base} with RL under randomized object geometries, varying scale, shape, and surface properties. The reward combines command tracking, whole-body control regularization, and the proposed interaction prior. Motion-reference tracking is removed deliberately *before* geometry-randomized RL, so that the policy learns to adapt to changed geometry through the same DF-conditioned interface it uses at deployment.

Table 2: **Object-scale generalization in simulation.** Column headers give object scale factors; $0.4\times$ and $1.6\times$ lie outside our per-axis post-training range $[0.6, 1.4]$. Ablations remove AIP, synthetic interaction data, geometry randomization, RL, or the Transformer. R_{succ} is the task success rate and R_{cont} the hand–object contact rate; entries are mean $\pm$ std over three seeds. Bold and underline mark the best and second-best results, * our implementation, † methods with one policy per task, and dashes settings that are not reported.

Method	PickUp R_{succ} (↑, %)					SitStand R_{succ} (↑, %)			Push R_{cont} (↑, %)			Carry R_{succ} (↑, %)
	0.4	0.6	1.0	1.4	1.6	0.4	1.0	1.6	0.6	1.0	1.4	0.4–1.6 Avg.
Motion-based baselines												
HDMI*	0.0 (±0.0)	<u>99.7</u> (±0.5)	**100.0** (±0.0)	40.7 (±3.3)	1.7 (±1.2)	0.0 (±0.0)	<u>99.0</u> (±0.0)	1.7 (±0.5)	0.3 (±0.0)	**97.3** (±0.5)	**10.6** (±2.9)	27.4 (±0.3)
ResMimic*	0.0 (±0.0)	18.3 (±1.2)	**100.0** (±0.0)	**99.7** (±0.5)	63.0 (±10.2)	0.0 (±0.0)	**100.0** (±0.0)	93.7 (±0.5)	0.4 (±0.2)	<u>93.1</u> (±1.2)	2.2 (±0.7)	32.9 (±1.5)
VisualMimic†	0.0 (±0.0)	0.0 (±0.0)	**100.0** (±0.0)	0.0 (±0.0)	0.0 (±0.0)	–	–	–	–	20.6 (±9.0)	–	–
Motion-free-at-inference baselines												
PhysHSI†	23.1 (±2.4)	70.5 (±2.6)	**100.0** (±0.0)	47.9 (±2.8)	39.9 (±2.8)	61.3 (±2.7)	76.2 (±2.3)	71.8 (±2.4)	–	–	–	<u>81.6</u> (±1.9)
Ablation study												
Ours - AIP	0.0 (±0.0)	0.7 (±0.5)	23.3 (±5.0)	57.3 (±1.2)	64.0 (±2.9)	**98.7** (±0.5)	98.3 (±0.5)	89.0 (±0.8)	**82.9** (±1.1)	72.5 (±1.9)	<u>6.5</u> (±1.9)	0.0 (±0.0)
Ours - Syn.	34.0 (±6.7)	94.3 (±2.5)	<u>99.3</u> (±0.9)	**99.7** (±0.5)	**99.7** (±0.5)	95.3 (±1.7)	30.0 (±2.2)	86.3 (±3.9)	0.0 (±0.0)	41.9 (±2.1)	3.5 (±1.6)	66.5 (±2.0)
Ours - Rand.	0.0 (±0.0)	0.0 (±0.0)	0.0 (±0.0)	0.0 (±0.0)	0.0 (±0.0)	0.0 (±0.0)	81.3 (±3.4)	**97.7** (±1.2)	0.0 (±0.0)	67.0 (±1.6)	2.1 (±1.0)	0.0 (±0.0)
Ours - RL	0.0 (±0.0)	2.0 (±0.8)	31.7 (±2.6)	37.0 (±2.4)	9.7 (±1.7)	1.7 (±1.2)	64.7 (±3.1)	15.0 (±2.8)	0.0 (±0.0)	67.3 (±0.7)	0.0 (±0.0)	5.3 (±1.5)
Ours - Trans.	0.0 (±0.0)	0.0 (±0.0)	0.0 (±0.0)	0.0 (±0.0)	0.0 (±0.0)	<u>98.0</u> (±0.8)	77.7 (±0.5)	<u>95.7</u> (±2.1)	<u>18.3</u> (±1.4)	85.7 (±2.4)	3.0 (±0.6)	0.0 (±0.0)
Ours (MoCap)	<u>63.0</u> (±5.0)	**100.0** (±0.0)	**100.0** (±0.0)	<u>88.0</u> (±1.6)	<u>94.0</u> (±1.6)	79.0 (±0.0)	80.3 (±2.1)	61.7 (±1.2)	13.9 (±0.2)	51.3 (±1.6)	2.0 (±0.3)	**82.9** (±1.4)
Ours (Vision)	**63.7** (±3.1)	94.7 (±1.9)	91.0 (±3.3)	**99.7** (±0.5)	93.0 (±1.6)	–	–	–	1.2 (±0.1)	35.2 (±1.3)	1.9 (±0.8)	35.8 (±2.7)

We introduce Adversarial Interaction Priors (AIP) as a cross-task interaction reward computed on velocity-augmented DF interaction statistics. It compares local robot–object interaction patterns instead of regularizing full-body kinematic trajectories, because valid interactions with different geometries may require different joint configurations yet should preserve similar proximity, approach, and surface-traversal behavior. For each queried link k, the link velocity $\mathbf{v}_{t,k}$ is decomposed in the local surface frame as

$$\mathbf{v}_{t,k}^{\text{norm}} = (\mathbf{v}_{t,k}^{\top}\mathbf{n}_{t,k})\mathbf{n}_{t,k}, \qquad \mathbf{v}_{t,k}^{\tan} = \mathbf{v}_{t,k} - \mathbf{v}_{t,k}^{\text{norm}}, \tag{5}$$

and the AIP discriminator receives

$$h_t = [z_t, \{\mathbf{v}_{t,k}^{\text{norm}}, \mathbf{v}_{t,k}^{\tan}\}_{k\in\mathcal{K}}], \tag{6}$$

where z_t is the DF-based latent used by the policy. The explicit velocities are needed because the temporal DF history captures changes in distance and surface direction but does not uniquely identify tangential motion; on a locally planar surface, static contact and sliding at different tangential velocities can produce nearly identical distance and normal histories. These velocities enter only the AIP discriminator during post-training; the deployed policy remains conditioned on z_t.

We build a reference buffer $\mathcal{B}_{\text{ref}}$ from rollouts of the mimic teacher π_{mimic} and train a discriminator D to separate its samples from policy samples with

$$\mathcal{L}_D = \mathbb{E}_{h\sim\mathcal{B}_{\text{ref}}}\left[(D(h)-1)^2\right] + \mathbb{E}_{h\sim\pi}\left[(D(h)+1)^2\right]. \tag{7}$$

The discriminator then defines the interaction reward

$$r_{\text{interact}}(h_t) = \max\left(0,\ 1-0.25(D(h_t)-1)^2\right). \tag{8}$$

The complete post-training reward is

$$r_t = r_{\text{cmd}} + \lambda_i r_{\text{interact}}(h_t) + \lambda_s r_{\text{style}} + r_{\text{reg}}, \tag{9}$$

where r_{cmd} tracks the root command, r_{style} regularizes motion naturalness, r_{reg} collects standard control and safety regularizers, and λ_i and λ_s weight the interaction and style rewards (Sec. A.3). Because the discriminator sees compact interaction statistics rather than the full robot state, AIP encourages geometrically valid interaction without enforcing a specific motion trajectory. Post-training yields the policy π_{full}.

Root commands and synthetic command augmentation. Post-training draws root-velocity commands from recorded and synthetic sources. For a recorded trajectory, the training command is the finite difference of the root trajectory projected onto the x–y plane. The 44 collected interaction clips contain limited locomotion diversity, so we augment them with synthetic samples that compose coarse root and object command primitives through predefined task templates. The primitives come from the same family used by the evaluation protocol, but the samples reuse none of the held-out composed evaluation sequences. Because inference is reference-free, the augmentation operates on commands and does not require synthesizing a new full-body motion trajectory for each command composition.

3.4 Visual-Motor Policy Distillation

The post-trained policy π_{full} computes DF queries from MoCap-derived object geometry, which is convenient for training and controlled evaluation but limits deployment in uninstrumented environments. To relax this sensing requirement, we distill π_{full} into a vision-based policy π_{vis} that observes egocentric depth.

The vision policy keeps the control interface of π_{full} but replaces the DF latent with an interaction latent predicted from depth,

$$o_{\text{vis}} = [o_{\text{prop}}, c_t^{\text{root}}, g_t, \hat{z}_t], \qquad \hat{z}_t = E_\phi(S_t), \tag{10}$$

where S_t is a history of egocentric depth frames and E_ϕ is a visual encoder trained to produce a compact interaction latent sufficient for the control policy. Visual distillation thus transfers the DF-based interface from privileged geometry to onboard perception.

Distillation is again DAgger-style, with the frozen π_{full} as the teacher. At each iteration, π_{vis} acts in simulation, π_{full} is queried at the visited states with its DF-based observation, and the student regresses the teacher's actions,

$$\mathcal{L}_{\text{distill}} = \mathbb{E}_{s\sim\pi_{\text{vis}}}\left[\left\|\pi_{\text{vis}}(o_{\text{vis}}) - \pi_{\text{full}}(o_{\text{base}})\right\|_2^2\right]. \tag{11}$$

We randomize camera extrinsics, depth noise, and physical parameters to improve robustness (Sec. A.4). Section 4.4 examines the performance of π_{vis} and the observability of contacts.

4 Experiments

The experiments test whether **LESSMIMIC** generalizes across object scale on all four skills and across shape on PickUp (Sec. 4.2), executes commanded multi-skill sequences without skill-specific policies (Sec. 4.3), and retains its performance when egocentric depth replaces the DF observation (Sec. 4.4). Component ablations (Sec. 4.5) and hardware experiments (Sec. 4.6) complete the evaluation.

4.1 Experimental Setup

Tasks and metrics. We evaluate four humanoid–object interaction tasks. PickUp requires lifting an object and holding it stably, SitStand requires stable pelvis–chair contact within a valid root-height range, and Carry requires transporting an object along a commanded trajectory; for these tasks we report the success rate R_{succ}. For Push, the single-task results report the hand–object contact rate R_{cont}, which measures contact maintenance rather than full pushing success. In sequential evaluation, every task instance, including Push, is judged by its maximum deviation from the commanded trajectory. Because the metrics differ across tasks, we report per-task results rather than one aggregated score (Sec. B.1).

Policy interface. At inference, the policy receives proprioception, the planar root-velocity command, the one-hot interaction-type flag, and either the DF latent or the depth-predicted latent, but no reference joint poses, future body motion, or contact trajectories. In sequential evaluation, the task order, root command, and interaction-type flag are specified externally at skill boundaries, and the policy must execute the commanded skills without environment resets or skill-specific policies.

Implementation and baselines. DF features are queried at the fixed link set of Sec. 3.1 and encoded over a short temporal history (Sec. A.1). We compare against the motion-conditioned baselines HDMI [50], ResMimic [62], and VisualMimic [56] and against the motion-free-at-inference baseline PhysHSI [47], all evaluated under the same task definitions and object-scale settings where applicable. Our post-training scales each object axis independently within $[0.6, 1.4]$, the same range used by our HDMI and ResMimic reproductions, so $0.4\times$ and $1.6\times$ are strict scale extrapolations. Each policy is evaluated on 128 episodes per task and scale (Sec. B.5).

4.2 Generalization Across Object Scales and Shapes

Changes in object geometry can substantially alter contact timing, reachability, and the required whole-body posture. We vary the object scale from $0.4\times$ to $1.6\times$ on all tasks (Push from $0.6\times$ to

Table 3: **Quantitative real-world deployment.** Success counts are over ten trials per setting. R_{acc} is the root-tracking accuracy, measured by external MoCap for evaluation only. Dashes mark SitStand, where the vision policy is not evaluated because its front camera poorly observes the back-side chair contact.

Method	**PickUp** 22 cm (↑, %)		**PickUp** 60 cm (↑, %)		**SitStand** 12 cm (↑, %)		**SitStand** 46 cm (↑, %)	
	Real R_{succ}	**Root** R_{acc}	**Real** R_{succ}	**Root** R_{acc}	**Real** R_{succ}	**Root** R_{acc}	**Real** R_{succ}	**Root** R_{acc}
Ours (MoCap)	10 / 10	94.44	8 / 10	81.39	8 / 10	84.89	10 / 10	91.88
Ours (Vision)	8 / 10	89.15	7 / 10	75.24	–	–	–	–

Table 4: **Matched geometric-representation ablation on PickUp.** Success rates (%) with the policy architecture, training pipeline, and scale-randomization protocol held fixed. Latent DF is the proposed temporally encoded representation; raw DF feeds the unencoded features to the policy but keeps the latent input for AIP. Bold marks the best value in each column.

Representation	**0.4**×	**0.6**×	**1.0**×	**1.4**×	**1.6**×	**Overall**
Latent DF	**63.0**	**100.0**	**100.0**	**88.0**	**94.0**	**89.0**
BBox	14.8	40.6	48.4	76.6	82.0	52.5
Voxel (16^3)	44.5	50.1	75.8	76.1	85.9	66.5
Raw DF	56.3	82.8	94.5	80.1	92.5	81.2

1.4×), with the nominal training scale at 1.0×. For PickUp, we also evaluate four held-out meshes (sphere, stool, cylinder, and torus) and the chair mesh, which appears in training only for other skills and therefore forms a new object–task pairing.

Motion-conditioned baselines often perform well near geometries compatible with their reference motions but degrade when the object scale changes substantially (Tab. 2). On PickUp, **LESSMIMIC** maintains strong performance across most scales, with at least 88.0% success at every scale except 0.4×, where its 63.0% still exceeds the 0.0% of all three motion-conditioned baselines. On Carry, the MoCap-based variant achieves the highest scale-averaged success among the compared methods (82.9% against 81.6% for PhysHSI), which suggests that the interaction latent is useful for sustained bimanual contact. The trends are more mixed for SitStand and Push, where some motion-conditioned baselines or ablated variants reach higher peak scores at particular scales, such as ResMimic on SitStand and HDMI on Push at 1.0×. Reference tracking provides a useful prior where the object geometry is compatible with the reference motion. The Push metric also differs, because it measures sustained hand contact, whereas PickUp and Carry measure completed object manipulation.

The matched ablation in Tab. 4 isolates the geometric input while keeping the rest of the training pipeline fixed. Raw DF already outperforms bounding-box and voxel observations overall (81.2% against 52.5% and 66.5%), and encoding the temporal DF history further raises success to 89.0%. Bounding-box and voxel success generally increases with object size. Larger objects offer a larger reachable region and tolerate less precise approach behavior, so this trend is consistent with representations that provide useful coarse geometry but encode the contact-relevant link-to-surface relation only indirectly. DF observations expose local distance and surface direction directly, which is especially useful for approaching and contacting small objects.

The learned latent is task-structured (Tab. A6). At the nominal scale, same-task cosine similarity ranges from 0.39 to 0.55, whereas latents of different tasks are nearly orthogonal; the exception is PickUp and Carry (0.41), which both involve sustained bimanual contact. Same-task similarity is lower across scales (0.08 to 0.28), so the latent is not strictly scale-invariant and preserves scale-dependent changes in interaction.

Shape transfer varies substantially across objects (Tab. A5). Scale-averaged PickUp success is 55.62% on the chair and 67.50% on the stool, against 11.25%, 10.94%, and 4.84% on the sphere, cylinder, and torus, where slippage and rolling before stable contact are common failures. Latent DF leads on four of the five meshes, whereas raw DF leads on the chair (61.09% against 55.62%). The policy also executes a second new object–task pairing, Push on the chair, in simulation and on hardware.

4.3 Sequential Multi-Skill Execution

Beyond isolated tasks, we evaluate whether the same deployed policy can execute externally commanded heterogeneous skills without resetting the robot or the environment. Each length-N sequence samples its task instances from the four task types and randomizes the object geometry, pose, and

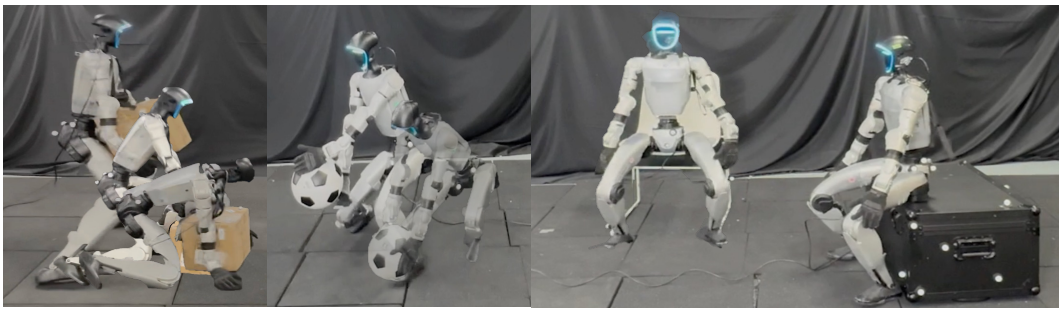


Figure 3: **Real-world generalization of LESSMIMIC.** The same policy (a) picks up a box from the training geometry family, (b) picks up a soccer ball, which is an unseen spherical object, and (c) performs SitStand on chairs of two heights (12 cm and 46 cm).

root command of each instance; composed sequences serve only for evaluation (Sec. B.3). At each skill boundary, the task script updates the interaction-type flag and the root command, and the policy continues from its current physical state. The protocol therefore tests whether one whole-body policy tolerates errors in balance, contact state, object placement, and root tracking that accumulate while it switches between commanded interaction modes. Only variants of the shared policy enter this benchmark, because reference-conditioned methods specify execution through motion and object trajectories and separate task policies would require a different skill-switching mechanism (Sec. B.4).

The MoCap-based policy succeeds on 62.1% of five-task sequences and on 23.5% of sequences with 15 task instances (Tab. 5). In a stress test that composes 40 instances of the same four task types under randomized configurations, success falls to 2.1% but remains nonzero. Success declines quickly with sequence length because a sequence succeeds only if *every* instance succeeds, and individual task success under the randomized distribution is below 100%. Ablated variants collapse after short sequences, which indicates that the full training stack is important for stable sequential execution.

4.4 Visual Distillation Analysis

The visual encoder is trained on depth observations rendered from three training meshes (a small box, a large case, and a chair) and never on the real-world foam roll or soccer ball used in the held-out deployment tests. The vision policy comes closest to the MoCap-based policy on PickUp, with success from 63.7% to 99.7% across scales (Tab. 2), but remains substantially weaker on Push, Carry, and sequential execution (Tabs. 2 and 5). This gap is expected, because π_{full} receives explicit DF queries from tracked object geometry, whereas π_{vis} must infer a policy-relevant interaction latent from egocentric depth, which is vulnerable to occlusion and a limited field of view.

The standard front camera poorly observes the back-side pelvis–chair contact in SitStand. To assess the role of visual coverage, we retrain the distillation with front and rear depth cameras that jointly cover the interaction. With both cameras, SitStand success reaches 0, 1, 31, 85, and 38% at 0.4, 0.6, 1.0, 1.4, and 1.6× (31% overall), against 0% at every scale with the front camera alone (Tab. A4). The improvement supports visual coverage as an important factor, and residual partial observability and latent-prediction error may contribute to the remaining gap to the privileged DF policy. When sequences involving SitStand are excluded, the standard vision policy succeeds on 36% of sequences, which suggests that the depth-predicted latent can support sequential execution when the relevant contacts are visible (Sec. C.3).

Table 5: **Sequential multi-skill execution in simulation.** Success rate (%) over N sequential task instances without resets, reported as mean ± std over three seeds. Bold marks the best result.

Method	N=5	N=10	N=15	N=25	N=40
Ablation study					
Ours - AIP	5.2 (±0.2)	0.0 (±0.0)	0.0 (±0.0)	0.0 (±0.0)	0.0 (±0.0)
Ours - Syn.	22.1 (±0.8)	4.9 (±0.2)	1.0 (±0.3)	0.0 (±0.0)	0.0 (±0.0)
Ours - Rand.	1.9 (±0.1)	0.0 (±0.0)	0.0 (±0.0)	0.0 (±0.0)	0.0 (±0.0)
Ours - RL	3.2 (±0.2)	0.0 (±0.0)	0.0 (±0.0)	0.0 (±0.0)	0.0 (±0.0)
Ours - Trans.	1.7 (±0.1)	0.0 (±0.0)	0.0 (±0.0)	0.0 (±0.0)	0.0 (±0.0)
Our method					
Ours (MoCap)	**62.1** (±1.7)	**38.1** (±1.6)	**23.5** (±1.2)	**9.0** (±0.6)	**2.1** (±0.2)
Ours (Vision)	15.9 (±0.6)	2.5 (±0.1)	0.0 (±0.0)	0.0 (±0.0)	0.0 (±0.0)

4.5 Ablation Study

Five ablations isolate the role of the major components. AIP is particularly important for Carry and sequential execution. Without it (*Ours - AIP*), the policy fails on Carry and succeeds on only 5.2% of five-task sequences and on none for $N \geqslant 10$. The same variant obtains higher Push contact rates

and higher SitStand success at all three reported scales. With the DF observation and the rest of the reward stack unchanged, removing the interaction prior lets the remaining rewards favor these contact metrics. The prior thus shapes the balance between individual contact objectives and coordinated multi-phase behavior (Sec. C.2).

The other ablations identify complementary sources of robustness. Removing synthetic data augmentation (*Ours - Syn.*) reduces performance on sustained contact and on sequential execution, where five-task success falls from 62.1% to 22.1%. These drops indicate that synthetic interaction data extends the coverage of object configurations and contact states beyond the collected demonstrations. Disabling geometry randomization (*Ours - Rand.*) degrades performance across several tasks and scales, and PickUp fails even at the nominal scale. Removing RL fine-tuning (*Ours - RL*) leaves the behavior-cloned policy with limited robustness under geometric variation. Replacing the Transformer with a Multi-Layer Perceptron (MLP) backbone (*Ours - Trans.*) degrades the tasks that require temporal coordination. This variant fails PickUp at every scale, fails Carry, and succeeds on only 1.7% of five-task sequences.

4.6 Real-World Deployment

We evaluate **LESSMIMIC** on a Unitree G1 with qualitative demonstrations and quantitative repeated trials. Qualitative tests cover all four skills and sequential execution (Figs. 3 and 4) and provide evidence that the same deployed policy interface transfers across the full task suite on hardware. Quantitative trials cover the two tasks with reliable and repeatable measurement protocols, PickUp across two object sizes and SitStand across two chair heights (Sec. B.6). External MoCap, used for evaluation only, measures the root-tracking accuracy R_{acc}, the fraction of frames with planar root-position error below $5\,\text{cm}$.

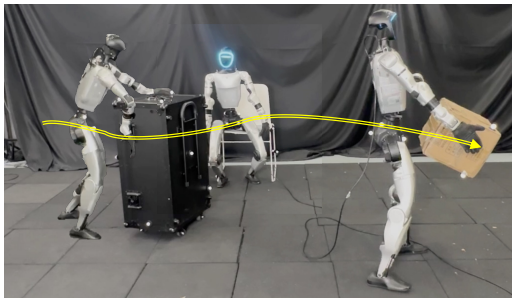


Figure 4: **Qualitative real-world sequential execution.** A single **LESSMIMIC** policy performs Push, PickUp, and Carry in sequence without resets, following externally provided task and root commands.

The MoCap-based policy transfers reliably to hardware in these settings, succeeding in 10 and 8 of 10 PickUp trials and in 8 and 10 of 10 SitStand trials (Tab. 3). The vision policy succeeds in both PickUp settings, in 8 and 7 of 10 trials, with lower success and tracking accuracy, consistent with the simulation gap from partial egocentric observability. It is not evaluated on SitStand because its camera poorly observes the back-side contact.

5 Conclusion

In **LESSMIMIC**, grounding control in local robot–object geometry rather than in deployed motion references lets one humanoid policy adapt to object-scale variation and new interaction settings, follow flexible commands, and compose heterogeneous skills in sequence. Unified DF representations and a three-stage pipeline of teacher-guided behavior cloning, AIP-guided RL, and visual-motor distillation turn this idea into policies that are motion-free at inference and deployable with either MoCap-based geometry or egocentric depth. In simulation, **LESSMIMIC** generalizes across four tasks and object scales from $0.4\times$ to $1.6\times$, succeeds on 62.1% of five-task sequences, and retains 23.5% success for 15 task instances. Hardware experiments show transfer across object geometries, with quantitative trials on PickUp and SitStand and qualitative demonstrations of the full task suite.

Three limitations remain (Sec. C.4). First, the per-link DF captures coarse proximity and surface direction rather than a detailed pressure distribution, so tasks that need precise force balance or fine contact placement remain difficult; force-aware feedback and denser geometric queries could help. Second, vision-based failures occur mainly when task-relevant contacts are occluded or outside the egocentric view, as for the back-side chair contact in SitStand; rear-facing or multi-view sensing could cover such contacts. Third, sensing latency can delay contact during fast motions, and the Python deployment stack adds roughly 5 to 15 ms of variable control latency; lower-latency perception and an optimized C++/ONNX control stack could reduce both.

Acknowledgments and Disclosure of Funding

This work is supported in part by the Brain Science and Brain-like Intelligence Technology—National Science and Technology Major Project (2025ZD0219400), the National Natural Science Foundation of China (62376009), the Beijing Nova program, the NVIDIA Academic Grant Program using Spark and Thor, the PKU-BingJi Joint Laboratory for Artificial Intelligence, the Wuhan Major Scientific and Technological Special Program (2025060902020304), the Hubei Embodied Intelligence Foundation Model Research and Development Program, and the National Comprehensive Experimental Base for Governance of Intelligent Society, Wuhan East Lake High-Tech Development Zone. The authors declare no competing interests.

References

[1] Allshire, A., Choi, H., Zhang, J., McAllister, D., Zhang, A., Kim, C. M., Darrell, T., Abbeel, P., Malik, J., and Kanazawa, A. (2025). Visual imitation enables contextual humanoid control. In *Conference on Robot Learning (CoRL)*.

[2] Chisari, E., Heppert, N., Argus, M., Welschehold, T., Brox, T., and Valada, A. (2024). Learning robotic manipulation policies from point clouds with conditional flow matching. In *Conference on Robot Learning (CoRL)*.

[3] Dai, Q., Zhu, Y., Geng, Y., Ruan, C., Zhang, J., and Wang, H. (2023). GraspNeRF: Multiview-based 6-DoF grasp detection for transparent and specular objects using generalizable NeRF. In *IEEE International Conference on Robotics and Automation (ICRA)*.

[4] Dietrich, A., Ott, C., and Albu-Schäffer, A. (2015). An overview of null space projections for redundant, torque-controlled robots. *International Journal of Robotics Research (IJRR)*, 34(11):1385–1400.

[5] Du, Y., Li, Y., Jia, B., Lin, Y., Zhou, P., Liang, W., Yang, Y., and Huang, S. (2026). COLA: Learning human-humanoid coordination for collaborative object carrying. In *IEEE International Conference on Robotics and Automation (ICRA)*.

[6] Fu, Y., Xie, F., Xu, C., Xiong, J., Yuan, H., and Lu, Z. (2026). DemoHLM: From one demonstration to generalizable humanoid loco-manipulation. *IEEE Robotics and Automation Letters (RA-L)*, 11(4):4393–4400.

[7] Fu, Z., Zhao, Q., Wu, Q., Wetzstein, G., and Finn, C. (2024). HumanPlus: Humanoid shadowing and imitation from humans. In *Conference on Robot Learning (CoRL)*.

[8] Haarnoja, T., Moran, B., Lever, G., Huang, S. H., Tirumala, D., Humplik, J., Wulfmeier, M., Tunyasuvunakool, S., Siegel, N. Y., Hafner, R., Bloesch, M., Hartikainen, K., Byravan, A., Hasenclever, L., Tassa, Y., Sadeghi, F., Batchelor, N., Casarini, F., Saliceti, S., Game, C., Sreendra, N., Patel, K., Gwira, M., Huber, A., Hurley, N., Nori, F., Hadsell, R., and Heess, N. (2024). Learning agile soccer skills for a bipedal robot with deep reinforcement learning. *Science Robotics*, 9(89):eadi8022.

[9] Hauser, K. (2016). Robust contact generation for robot simulation with unstructured meshes. In *International Symposium on Robotics Research*.

[10] He, T., Luo, Z., He, X., Xiao, W., Zhang, C., Zhang, W., Kitani, K., Liu, C., and Shi, G. (2024a). OmniH2O: Universal and dexterous human-to-humanoid whole-body teleoperation and learning. In *Conference on Robot Learning (CoRL)*.

[11] He, T., Luo, Z., Xiao, W., Zhang, C., Kitani, K., Liu, C., and Shi, G. (2024b). Learning human-to-humanoid real-time whole-body teleoperation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

[12] Herzog, A., Rotella, N., Mason, S., Grimminger, F., Schaal, S., and Righetti, L. (2016). Momentum control with hierarchical inverse dynamics on a torque-controlled humanoid. *Autonomous Robots*, 40(3):473–491.

[13] Hopkins, M. A., Leonessa, A., Lattimer, B. Y., and Hong, D. W. (2016). Optimization-based whole-body control of a series elastic humanoid robot. *International Journal of Humanoid Robotics*, 13(01):1550034.

[14] Huang, W., Wang, C., Zhang, R., Li, Y., Wu, J., and Fei-Fei, L. (2023). VoxPoser: Composable 3D value maps for robotic manipulation with language models. In *Conference on Robot Learning (CoRL)*.

[15] Ji, M., Peng, X., Liu, F., Li, J., Yang, G., Cheng, X., and Wang, X. (2026). ExBody2: Advanced expressive humanoid whole-body control. In *IEEE International Conference on Robotics and Automation (ICRA)*.

[16] Jia, X., Wang, Q., Wang, A., Wang, H. A., Gyenes, B., Gospodinov, E., Jiang, X., Li, G., Zhou, H., Liao, W., Huang, X., Beck, M., Reuss, M., Lioutikov, R., and Neumann, G. (2025). PointMapPolicy: Structured point cloud processing for multi-modal imitation learning. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*.

[17] Jiang, Z., Zhu, Y., Svetlik, M., Fang, K., and Zhu, Y. (2021). Synergies between affordance and geometry: 6-DoF grasp detection via implicit representations. In *Robotics: Science and Systems (RSS)*.

[18] Jing, Z., Yang, S., Ao, J., Xiao, T., Jiang, Y.-G., and Bai, C. (2025). HumanoidGen: Data generation for bimanual dexterous manipulation via LLM reasoning. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*.

[19] Khatib, O., Sentis, L., Park, J., and Warren, J. (2004). Whole-body dynamic behavior and control of human-like robots. *International Journal of Humanoid Robotics*, 1(01):29–43.

[20] Kumar, A., Fu, Z., Pathak, D., and Malik, J. (2021). RMA: Rapid motor adaptation for legged robots. In *Robotics: Science and Systems (RSS)*.

[21] Li, D., Chen, X., Wu, Q., Chen, B., Wu, S., Wu, H., Zhang, G., Li, L., Zhou, M., Xiang, D., Ma, J., Zhang, Q., and Xu, R. (2026a). HAIC: Humanoid agile object interaction control via dynamics-aware world model. In *Robotics: Science and Systems (RSS)*.

[22] Li, J., Cheng, X., Huang, T., Yang, S., Qiu, R.-Z., and Wang, X. (2025a). AMO: Adaptive motion optimization for hyper-dexterous humanoid whole-body control. In *Robotics: Science and Systems (RSS)*.

[23] Li, Y., Lin, Y., Cui, J., Liu, T., Liang, W., Zhu, Y., and Huang, S. (2025b). CLONE: Closed-loop whole-body humanoid teleoperation for long-horizon tasks. In *Conference on Robot Learning (CoRL)*.

[24] Li, Y., Luo, Z., Zhang, T., Dai, C., Kanervisto, A., Tirinzoni, A., Weng, H., Kitani, K., Guzek, M., Touati, A., Lazaric, A., Pirotta, M., and Shi, G. (2026b). BFM-Zero: A promptable behavioral foundation model for humanoid control using unsupervised reinforcement learning. In *Proceedings of International Conference on Learning Representations (ICLR)*.

[25] Li, Y., Zhang, Y., Razmjoo, A., and Calinon, S. (2024). Representing robot geometry as distance fields: Applications to whole-body manipulation. In *IEEE International Conference on Robotics and Automation (ICRA)*.

[26] Li, Y., Zhang, Y., Xiao, W., Pan, C., Weng, H., He, G., He, T., and Shi, G. (2025c). Hold my beer: Learning gentle humanoid locomotion and end-effector stabilization control. In *Conference on Robot Learning (CoRL)*.

[27] Liao, Q., Truong, T. E., Huang, X., Gao, Y., Tevet, G., Sreenath, K., and Liu, C. K. (2026). BeyondMimic: From motion tracking to versatile humanoid control via guided diffusion. *Science Robotics*, 11(117):eadx8924.

[28] Liu, M., Li, X., Ling, Z., Li, Y., and Su, H. (2022). Frame mining: a free lunch for learning robotic manipulation from 3D point clouds. In *Conference on Robot Learning (CoRL)*.

[29] Liu, W., Wan, Y., Wang, J., Kuang, Y., Shi, X., Li, H., Zhao, D., Zhang, Z., and Wang, H. (2025). FetchBot: Learning generalizable object fetching in cluttered scenes via zero-shot Sim2Real. In *Conference on Robot Learning (CoRL)*.

[30] Lunardi, G., Corbères, T., Mastalli, C., Mansard, N., Flayols, T., Tonneau, S., and Del Prete, A. (2024). Reference-free model predictive control for quadrupedal locomotion. *IEEE Access*, 12:689–698.

[31] Luo, Z., Yuan, Y., Wang, T., Li, C., Castañeda, F., Chen, S., Cao, Z.-A., Li, J., Minor, D., Ben, Q., Park, J., Sami, D., Wang, Z., Da, X., Ding, R., Hogg, C., Song, L., Lim, E., Jeong, E., He, T., Xue, H., Xiao, W., Yuen, S., Kautz, J., Chang, Y., Iqbal, U., Fan, L., and Zhu, Y. (2026). SONIC: Supersizing motion tracking for natural humanoid whole-body control. *Science Robotics*, 11(117):eaed4592.

[32] Moro, F. L. and Sentis, L. (2019). Whole-body control of humanoid robots. In *Humanoid Robotics: a Reference*, pages 1161–1183. Springer Dordrecht.

[33] Nakamura, Y., Hanafusa, H., and Yoshikawa, T. (1987). Task-priority based redundancy control of robot manipulators. *International Journal of Robotics Research (IJRR)*, 6(2):3–15.

[34] Osher, S. and Sethian, J. A. (1988). Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 79(1):12–49.

[35] Pan, J., Chitta, S., and Manocha, D. (2012). FCL: A general purpose library for collision and proximity queries. In *IEEE International Conference on Robotics and Automation (ICRA)*.

[36] Peng, X. B., Ma, Z., Abbeel, P., Levine, S., and Kanazawa, A. (2021). AMP: Adversarial motion priors for stylized physics-based character control. *ACM Transactions on Graphics (TOG)*, 40(4):1–20.

[37] Posa, M. and Tedrake, R. (2013). Direct trajectory optimization of rigid body dynamical systems through contact. In *Tenth Workshop on the Algorithmic Foundations of Robotics*.

[38] Quintero-Pena, C., Thomason, W., Kingston, Z., Kyrillidis, A., and Kavraki, L. E. (2024). Stochastic implicit neural signed distance functions for safe motion planning under sensing uncertainty. In *IEEE International Conference on Robotics and Automation (ICRA)*.

[39] Ross, S., Gordon, G., and Bagnell, D. (2011). A reduction of imitation learning and structured prediction to no-regret online learning. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*.

[40] Schulman, J., Duan, Y., Ho, J., Lee, A., Awwal, I., Bradlow, H., Pan, J., Patil, S., Goldberg, K., and Abbeel, P. (2014). Motion planning with sequential convex optimization and convex collision checking. *International Journal of Robotics Research (IJRR)*, 33(9):1251–1270.

[41] Sentis, L. and Khatib, O. (2005). Synthesis of whole-body behaviors through hierarchical control of behavioral primitives. *International Journal of Humanoid Robotics*, 2(04):505–518.

[42] Shen, B., Xia, F., Li, C., Martín-Martín, R., Fan, L., Wang, G., Pérez-D'Arpino, C., Buch, S., Srivastava, S., Tchapmi, L. P., Tchapmi, M. E., Vainio, K., Wong, J., Li, F.-F., and Savarese, S. (2021). iGibson 1.0: A simulation environment for interactive tasks in large realistic scenes. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7520–7527.

[43] Shi, J., Liu, X., Wang, D., Lu, O., Schwertfeger, S., Zhang, C., Sun, F., Bai, C., and Li, X. (2025). Adversarial locomotion and motion imitation for humanoid policy learning. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*.

[44] Song, M., Ha, J., Park, B., and Park, D. (2025). Implicit neural-representation learning for elastic deformable-object manipulations. In *Robotics: Science and Systems (RSS)*.

[45] Tessler, C., Kasten, Y., Guo, Y., Mannor, S., Chechik, G., and Peng, X. B. (2023). CALM: Conditional adversarial latent models for directable virtual characters. In *ACM SIGGRAPH Conference Proceedings*.

[46] Varley, J., DeChant, C., Richardson, A., Ruales, J., and Allen, P. (2017). Shape completion enabled robotic grasping. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

[47] Wang, H., Zhang, W., Yu, R., Huang, T., Ren, J., Jia, F., Wang, Z., Niu, X., Chen, X., Chen, J., Chen, Q., Wang, J., and Pang, J. (2025a). PhysHSI: Towards a real-world generalizable and natural humanoid-scene interaction system. *arXiv preprint arXiv:2510.11072*.

[48] Wang, Y., Yang, M., Ding, Z., Zhang, Y., Zeng, W., Xu, X., Jiang, H., and Lu, Z. (2025b). From experts to a generalist: Toward general whole-body control for humanoid robots. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*.

[49] Wang, Y., Zhao, Q., Lau, Y. F., Yu, R., Tsui, H. W., Chen, Q., Wang, J., Pang, J., and Tan, P. (2026). HumanX: Toward agile and generalizable humanoid interaction skills from human videos. In *Conference on Robot Learning (CoRL)*.

[50] Weng, H., Li, Y., Sobanbabu, N., Wang, Z., Luo, Z., He, T., Ramanan, D., and Shi, G. (2025). HDMI: Learning interactive humanoid whole-body control from human videos. *arXiv preprint arXiv:2509.16757*.

[51] Xie, W., Han, J., Zheng, J., Li, H., Liu, X., Shi, J., Zhang, W., Bai, C., and Li, X. (2025). KungfuBot: Physics-based humanoid whole-body control for learning highly-dynamic skills. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*.

[52] Xu, Z., He, Z., Wu, J., and Song, S. (2020). Learning 3D dynamic scene representations for robot manipulation. In *Conference on Robot Learning (CoRL)*.

[53] Xue, H., Huang, X., Niu, D., Liao, Q., Kragerud, T., Gravdahl, J. T., Peng, X. B., Shi, G., Darrell, T., Sreenath, K., and Sastry, S. (2025). LeVERB: Humanoid whole-body control with latent vision-language instruction. *arXiv preprint arXiv:2506.13751*.

[54] Yang, L., Huang, X., Wu, Z., Kanazawa, A., Abbeel, P., Sferrazza, C., Liu, C. K., Duan, R., and Shi, G. (2026). OmniRetarget: Interaction-preserving data generation for humanoid whole-body loco-manipulation and scene interaction. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 11691–11698.

[55] Yang, W. and Jin, W. (2025). ContactSDF: Signed distance functions as multi-contact models for dexterous manipulation. *IEEE Robotics and Automation Letters (RA-L)*, 10(5):4212–4219.

[56] Yin, S., Ze, Y., Yu, H.-X., Liu, C. K., and Wu, J. (2025). VisualMimic: Visual humanoid loco-manipulation via motion tracking and generation. *arXiv preprint arXiv:2509.20322*.

[57] Ze, Y., Chen, Z., Araújo, J. P., Cao, Z.-a., Peng, X. B., Wu, J., and Liu, C. K. (2025). TWIST: Teleoperated whole-body imitation system. In *Conference on Robot Learning (CoRL)*.

[58] Ze, Y., Zhao, S., Wang, W., Kanazawa, A., Duan, R., Abbeel, P., Shi, G., Wu, J., and Liu, C. K. (2026). TWIST2: Scalable, portable, and holistic humanoid data collection system. In *IEEE International Conference on Robotics and Automation (ICRA)*.

[59] Zesch, R. S., Modi, V., Sueda, S., and Levin, D. I. (2023). Neural collision fields for triangle primitives. In *ACM SIGGRAPH Asia Conference Proceedings*.

[60] Zhang, X., Wang, C., Sun, L., Wu, Z., Zhu, X., and Tomizuka, M. (2023). Efficient sim-to-real transfer of contact-rich manipulation skills with online admittance residual learning. In *Conference on Robot Learning (CoRL)*.

[61] Zhang, Z., Guo, J., Chen, C., Wang, J., Lin, C., Lian, Y., Xue, H., Wang, Z., Liu, M., Lyu, J., Liu, H., Wang, H., and Yi, L. (2026). Track any motions under any disturbances. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 17533–17540.

[62] Zhao, S., Ze, Y., Wang, Y., Liu, C. K., Abbeel, P., Shi, G., and Duan, R. (2026). ResMimic: From general motion tracking to humanoid whole-body loco-manipulation via residual learning. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*.

[63] Zheng, C., Chen, K., Bi, Z., Li, Y., Pan, L., Zhou, J., Li, H., and Ma, J. (2026). Embracing bulky objects with humanoid robots: Whole-body manipulation with reinforcement learning. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 6089–6096.

[64] Zhu, H., Wang, Y., Huang, D., Ye, W., Ouyang, W., and He, T. (2024). Point cloud matters: Rethinking the impact of different observation spaces on robot learning. In *Proceedings of the Neural Information Processing Systems (NeurIPS) Track on Datasets and Benchmarks*.

A Implementation Details

This section details the interaction representation, policy architecture, training stages, reward formulation, and visual-motor distillation of the **LESSMIMIC** pipeline introduced in Sec. 3. Unless stated otherwise, all policy variants share the same whole-body control architecture and differ only in the sensing input or the training objective.

A.1 Interaction Representation and Encoder

We compute distance-field queries from the object geometry with NVIDIA Kaolin. In the real-world setup with Motion Capture (MoCap), the object mesh and rigid pose are known, so no online mesh reconstruction is required; the queried humanoid links are transformed into the object frame before a batched point-to-surface evaluation. On an RTX 5090, the measured latency stays below $9\,\mathrm{ms}$ per pass even for 1024 parallel environments with three objects (Tab. A1). Queries are evaluated at a fixed set of humanoid links shared across all skills, namely both wrists, both ankles, and the pelvis. The fixed query set avoids task-specific observation layouts and keeps the policy interface identical across PickUp, SitStand, Push, and Carry.

For each queried link, the deployable geometric observation contains the distance to the nearest object surface and a local surface-direction estimate. At differentiable points of the distance field, the direction is the normalized gradient. At non-smooth regions, such as sharp edges, corners, or medial-axis-like regions where the nearest surface projection is not unique, we use a normalized numerical direction estimate instead. When this estimate changes abruptly, we retain the last valid direction and rely on the encoder of the Variational Auto-Encoder (VAE) to smooth the resulting local discontinuity before the policy consumes the feature.

The interaction history spans a temporal window of length $l = 5$ and is encoded into a 64-dimensional latent z_t. Although the raw feature is already compact, the VAE does more than reduce dimensionality. It smooths discontinuous local geometry cues, regularizes the interaction representation, and provides a stable latent space for policy conditioning and adversarial interaction regularization.

The encoder predicts a Gaussian posterior

$$q_\phi(z_t \mid I_t) = \mathcal{N}(\mu_\phi(I_t), \sigma_\phi(I_t)), \tag{A1}$$

the decoder reconstructs the interaction history, and the VAE is trained with

$$\mathcal{L}_{\mathrm{VAE}} = \|\hat{I}_t - I_t\|_2^2 + \beta D_{\mathrm{KL}}\left(q_\phi(z_t \mid I_t) \,\|\, \mathcal{N}(0, I)\right). \tag{A2}$$

The VAE is trained jointly with the policy during interaction skill pre-training under the DAgger behavior-cloning objective, so gradients from both the reconstruction objective and policy learning update the interaction encoder. After pre-training, the encoder is frozen and produces z_t during discriminative post-training. During visual-motor distillation, the frozen Distance Field (DF) encoder defines the teacher-side latent, while the visual encoder learns to predict a policy-relevant replacement latent from egocentric depth.

During discriminative post-training with Adversarial Interaction Priors (AIP), we augment the DF latent with link velocities decomposed in the local surface frame into a normal component along the local DF direction and a tangential remainder (Sec. 3.3). The deployed policy thus remains conditioned on z_t without these velocities, whereas the AIP discriminator receives the velocity-augmented statistic $h_t = [z_t, \{\mathbf{v}_{t,k}^{\mathrm{norm}}, \mathbf{v}_{t,k}^{\tan}\}_{k\in\mathcal{K}}]$ during post-training.

Table A1: **Latency of distance-field queries on one RTX 5090.** Time per pass (ms) for 1, 7, and 30 queries.

Environments	Objects	1 query	7 queries	30 queries
1	1	3.449	3.260	3.313
1	2	6.135	5.832	6.002
1	3	8.784	8.898	8.712
1024	1	3.230	3.283	3.278
1024	2	6.000	6.007	6.043
1024	3	8.720	8.808	8.776

Table A2: **Training hyperparameters of the three stages.** The stages share one policy architecture and differ in their optimization settings.

Hyperparameter	Symbol	Value
Interaction Skill Pre-Training		
Learning rate	η_{pre}	1×10^{-3}
Batch size	B_{pre}	4096×8
Number of training iterations	N_{pre}	24,000
Optimizer	–	Adam
Discriminative Post-Training		
Policy learning rate	η_{pol}	1×10^{-3}
Discriminator learning rate	η_{disc}	2×10^{-4}
Discount factor	γ	0.99
GAE lambda	λ	0.95
Entropy coefficient	α_{ent}	5×10^{-4}
Number of environments	N_{env}	4096×8
Number of environment steps	N_{rl}	240,000
Visual-Motor Distillation		
Learning rate	η_{dist}	1×10^{-3}
Batch size	B_{dist}	2048×8
Number of distillation iterations	N_{dist}	120,000

A.2 Policy Architecture and Training Stages

Teacher–student pre-training. We train one multi-task mimic teacher π_{mimic} following ResMimic [62]. The teacher tracks 44 retargeted human–object interaction motions that span PickUp, SitStand, Push, and Carry and are collected with three training object instances: a box for PickUp, a chair for SitStand, and a large box for Push. It conditions on the current reference-motion state together with privileged body velocity, object pose, and contact information. Parallel simulation environments pair each object with its reference motion or motions, and their state distributions are combined to train this one teacher. The student policy π_{base} is trained with DAgger-style behavior cloning on the deployment observation o_{base} of Eq. (3), in which c_t^{root} is the commanded planar floating-base velocity. The student receives no reference joint poses, future body motion, or contact trajectories, whereas the teacher input o_{mimic} in Eq. (4) contains privileged full-body reference-motion information. This asymmetry lets motion-imitation supervision initialize physically feasible contact behavior while the policy interface remains motion-free at inference. All task and object rollouts are sampled jointly to update the same π_{base}. This stage uses no scale or geometry randomization, which enters only in discriminative post-training.

Policy architecture. All policy variants use the same Transformer-based whole-body controller. Its input tokens encode proprioception, the root-velocity command, the interaction-type flag, and the interaction latent; its output is the whole-body action sent to the low-level controller. Pre-training, post-training, and visual distillation share this architecture and change only the training objective and the sensing modality.

Root-command construction and synthetic augmentation. Post-training draws root-velocity commands from recorded and synthetic sources. Recorded commands are finite differences of the reference root trajectory projected onto the x–y plane. To supplement the limited locomotion diversity of the 44 collected interaction clips, synthetic samples compose coarse root and object command primitives through predefined task templates. These primitives match the command family used in evaluation, but no held-out composed evaluation sequence is reused.

A.3 Discriminative Post-Training Rewards

Post-training combines root-velocity tracking, AIP interaction regularization, a whole-body motion-style prior, and control penalties in the reward r_t of Eq. (9), with weights shared across tasks (Tab. A3). The interaction reward in Eq. (8) takes the least-squares form of Adversarial Motion Priors (AMP) [36] and is evaluated on the velocity-augmented interaction statistic h_t. A separate discriminator over

Table A3: **Reward terms for discriminative post-training.** In the formulations, $\mathbf{v}_t^{\text{root},xy}$ is the measured planar root velocity, c_t^{root} its command, and $D(s)$ the motion-style discriminator over robot states.

Reward Term	Formulation	Weight	Description
Task and Style Rewards			
Root Velocity Tracking	$-\lVert \mathbf{v}_t^{\text{root},xy} - c_t^{\text{root}} \rVert^2$	1.0	Track the commanded planar root velocity without motion references.
Interaction Style	Eq. (8)	2.0	Discriminator-based prior on the distance-field interaction representation.
Motion Style	$1 - 0.25(D(s) - 1)^2$	1.0	Discriminator-based motion prior for natural gait and posture.
Regularization and Penalties			
Action Reg.	$-\lVert \Delta a_t \rVert^2$	5.0	Penalize abrupt action changes to improve control stability.
Termination	Constant	-10.0	Applied upon early termination, such as falls or loss of balance.
Joint Limit	n_{exceed}	-5.0	Penalize joints exceeding specified soft limits.

Table A4: **SitStand success (%) across object scales with front-only and front–rear simulated depth.** A rear depth view exposes the pelvis–chair contact that the standard front-facing camera poorly observes.

Depth model	0.4×	0.6×	1.0×	1.4×	1.6×	Overall
Front-only	0	0	0	0	0	0
Front–rear	0	1	31	85	38	31

robot states supplies the motion-style reward. Control penalties cover action changes, joint limits, and early termination.

A.4 Visual-Motor Distillation Details

The vision policy replaces the privileged MoCap-derived distance-field observation with egocentric depth and observes o_{vis} in Eq. (10), where the visual encoder E_ϕ is trained to predict, from the history of egocentric depth frames S_t, a compact interaction latent sufficient for the control policy. Distillation is DAgger-style, with the frozen MoCap-based teacher π_{full}. At the visited states, the teacher is queried with its distance-field observation, and the student minimizes the action regression $\mathcal{L}_{\text{distill}}$ in Eq. (11).

We apply domain randomization to camera extrinsics, depth noise, object physical parameters, and robot dynamics to improve robustness. The visual encoder is trained on depth observations from three training meshes (a small box, a large box, and a chair) and never on the real-world foam roll or soccer ball used for the held-out deployment tests.

B Experimental Details

B.1 Tasks, Metrics, and Success Criteria

The four skills involve different contact objectives, so each task has its own metric.

PickUp. A trial succeeds if the object is lifted above the height threshold and remains stably held for the required duration.

SitStand. A trial succeeds if the robot establishes stable pelvis–chair contact while keeping the root height within the valid range. This metric evaluates seated contact and posture stability rather than the precise reproduction of a reference motion.

Push. The hand–object contact rate R_{cont} is measured over the task horizon, and body-dominated contact is not counted. The metric thus measures whether the robot maintains the intended bimanual contact with the pushed object.

Carry and sequential multi-skill execution. Carry and sequential multi-skill execution are both judged by whether the manipulated object follows the commanded trajectory within a fixed tolerance. Sequential success requires completing all task instances in the sequence without resetting the robot or the environment.

B.2 Compute Resources

All training is conducted on eight NVIDIA RTX 5090 GPUs. Training the motion-tracking mimic teacher takes approximately 24 hours, interaction skill pre-training of π_{base} approximately 1 hour, discriminative post-training with AIP-guided Reinforcement Learning (RL) approximately 24 hours, and visual-motor distillation approximately 4 hours. These estimates correspond to the training configurations in Tab. A2; data preprocessing, evaluation rollouts, and real-world deployment calibration may require additional wall-clock time.

B.3 Sequential Multi-Skill Protocol

Sequential evaluation tests the temporally extended execution of one shared low-level policy. The task sequence is specified externally, and the one-hot interaction-type flag g_t provides the active interaction mode.

Evaluation sequence generation. Command sequences are synthesized only for evaluation and never serve as additional training demonstrations. For each sequence length N, we randomly sample N task instances from PickUp, SitStand, Push, and Carry. Each task type has predefined command templates and transition rules that specify the interaction-type flag, the nominal task phase, and the condition for switching to the next task. Within each sampled task instance, the object geometry, object pose, and root-velocity command are randomized according to the task distribution. The benchmark therefore evaluates execution under randomized commanded task sequences rather than imitation of a prerecorded long-horizon motion reference.

Execution and success. During training, skill labels come from annotations of the 44 collected interaction trajectories, but the policy is never trained on full composed multi-task sequences. At evaluation, the operator or task script updates the one-hot interaction-type flag at task boundaries and provides joystick or scripted root commands, which specify the desired planar velocity of the floating base. The policy must execute each commanded skill from its current physical state, without environment resets or switching to a skill-specific policy, and a sequence counts as successful only if all N task instances are completed.

This protocol is challenging because physical errors accumulate across skills. Small root-tracking drift, object displacement, unstable contact, or balance errors can shift the state distribution before the next skill begins. The sequential evaluation therefore measures the robustness of a shared low-level interaction policy under accumulated state drift.

B.4 Baselines and Ablations

We compare against representative motion-conditioned and motion-free-at-inference baselines, with matched task definitions and scale settings wherever their required observations and reference inputs are available. Motion-conditioned baselines require motion or pose references at inference, whereas **LESSMIMIC** uses a root command, an interaction-type flag, and geometric interaction observations. The full model, HDMI, and ResMimic share the per-axis scale-randomization interval $[0.6, 1.4]$, so evaluations at $0.4\times$ and $1.6\times$ test extrapolation beyond it.

The sequential benchmark compares ablations of the same deployed policy. HDMI and ResMimic accept motion and object references, including the transitions needed for a composed sequence; PhysHSI and VisualMimic use separate policies for individual skills. These differences in input and policy structure limit direct comparison under the interaction-type flag and root-command protocol used here.

The ablations in the main paper isolate five components.

- **Ours - AIP** removes the adversarial interaction reward and the discriminator update, while keeping the distance-field policy input and the rest of the RL reward stack unchanged.

Table A5: **PickUp success (%) on additional object geometries and an untrained object–task pairing.** The sphere, stool, cylinder, and torus meshes are held out from training; the chair mesh is seen in training for other interactions but not for PickUp. Each row reports success at scales from $0.4\times$ to $1.6\times$ and the mean over scales.

Representation	Geometry	0.4	0.6	1.0	1.4	1.6	Overall
Latent DF	Chair	63.28	95.31	99.22	18.75	1.56	55.62
Latent DF	Sphere	54.69	1.56	0.00	0.00	0.00	11.25
Latent DF	Stool	48.44	99.22	100.00	42.19	47.66	67.50
Latent DF	Cylinder	0.78	39.84	14.06	0.00	0.00	10.94
Latent DF	Torus	0.00	0.00	22.66	0.00	1.56	4.84
BBox	Chair	16.41	17.97	51.56	21.88	4.69	22.50
BBox	Sphere	0.00	0.78	0.00	0.00	0.00	0.16
BBox	Stool	46.09	27.34	7.81	2.34	1.56	17.03
BBox	Cylinder	1.56	12.50	2.34	0.00	0.00	3.28
BBox	Torus	0.00	0.00	1.56	0.00	3.91	1.09
Voxel	Chair	68.75	96.88	96.88	21.88	2.34	57.34
Voxel	Sphere	0.00	0.00	0.00	0.00	0.00	0.00
Voxel	Stool	46.09	27.34	91.41	14.84	5.47	37.03
Voxel	Cylinder	0.78	17.19	0.00	0.00	0.00	3.59
Voxel	Torus	0.00	0.00	3.12	3.91	0.00	1.41
Raw DF	Chair	92.97	91.41	92.97	18.75	9.38	61.09
Raw DF	Sphere	3.91	0.00	0.00	0.00	0.00	0.78
Raw DF	Stool	78.91	70.31	89.06	27.34	21.88	57.50
Raw DF	Cylinder	0.00	27.34	0.00	0.00	0.00	5.47
Raw DF	Torus	0.00	0.00	9.38	3.91	0.00	2.66

- **Ours - Syn.** removes the synthetic interaction data augmentation.
- **Ours - Rand.** disables geometry randomization.
- **Ours - RL** removes RL fine-tuning after behavior cloning.
- **Ours - Trans.** replaces the Transformer backbone with a Multi-Layer Perceptron (MLP) backbone.

B.5 Simulation Evaluation Protocol

For every simulation result, including the representation ablation, the additional-geometry evaluation, and the front–rear depth ablation, we evaluate each trained policy over 128 episodes per task and scale. Results are averaged over three seeds, and standard deviations, where reported, are computed over seed-level averages. The mean root-position error across the evaluated simulation motions is $0.1298\,\mathrm{m}$. Sequential trials use the all-instance completion criterion of Sec. B.3. Dashes in the result tables mark settings unsupported by the original method interface or unavailable under the required observation and reference inputs.

B.6 Real-World Evaluation Protocol

Real-world evaluation has qualitative and quantitative components. Qualitatively, we test all four skills (PickUp, SitStand, Push, and Carry) and sequential multi-skill execution. Quantitatively, we report ten repeated trials per setting for PickUp and SitStand, because these tasks have reliable and repeatable measurement protocols on hardware; Push, Carry, and sequential execution are reported qualitatively. External MoCap measures root-tracking accuracy for evaluation only. The vision policy is not evaluated quantitatively on SitStand, because the egocentric camera poorly observes the key back-side chair contact.

B.7 Geometry, Object–Task Transfer, and Representation Evaluation

Table A5 compares four representations under the same training protocol on five PickUp meshes, four held out from training and one paired with a new task, at five scales. Latent DF has the highest scale-averaged success on four meshes and raw DF on the chair. Highly curved objects remain difficult for all representations; the dominant failures are hand–object slippage and, for spherical objects, rolling before stable contact is established.

We also tested a second new object–task pairing, the Push command on the chair, and the policy pushed the chair in simulation and on hardware. Hardware attempts to pick up the chair were limited by motor overheating under the chair's weight.

Table A6: **Cosine similarity of interaction latents across tasks and scales.** Each trajectory-level latent is the VAE encoder mean averaged over a 5-second window around the main contact event. Diagonal entries report same-task similarity at the nominal scale $1.0\times$, values in parentheses same-task similarity averaged over all pairs of different scales, and off-diagonal entries cross-task similarity at $1.0\times$.

	Carry	PickUp	Push	SitStand
Carry	0.51 (0.24)	0.41	-0.02	-0.04
PickUp	0.41	0.55 (0.28)	-0.03	-0.04
Push	-0.02	-0.03	0.39 (0.08)	-0.06
SitStand	-0.04	-0.04	-0.06	0.48 (0.22)

C Additional Analysis

C.1 Representation Similarity Across Tasks and Scales

The latent similarities in Tab. A6 use the VAE encoder mean $\mu_\phi(I_t)$ as the interaction latent instead of a sample from the posterior. Latents are collected from evaluation rollouts of the full **LESSMIMIC** policy, with ten rollout trajectories for each task–scale pair. For each trajectory, we extract the segment within a 5-second window around the main contact event, average its latent vectors into one trajectory-level interaction representation, and ℓ_2-normalize this representation before computing cosine similarities.

Each same-scale diagonal entry averages the pairwise cosine similarities between different trajectories of the same task at the nominal scale $1.0\times$, excluding self-pairs. Each value in parentheses averages the pairwise similarities between trajectory-level representations of the same task over all pairs of different scales. Off-diagonal entries average the pairwise similarities between trajectory-level representations of different tasks at the nominal scale $1.0\times$. Each same-scale entry thus uses ten trajectories per task at the nominal scale, and each cross-scale entry uses ten trajectories for every task–scale pair.

Interaction latents retain higher intra-task similarity than unrelated inter-task pairs, and PickUp and Carry share a relatively high mutual similarity because both involve sustained bimanual object interaction. Same-task similarity is lower across scales than at the nominal scale for every task, and Push shows the weakest cross-scale similarity, which suggests that pushing behavior is more sensitive to object scale and contact timing than PickUp or Carry. The latent is therefore not strictly scale-invariant; it remains task-structured while preserving scale-dependent changes in interaction.

C.2 AIP Trade-Off Analysis

Removing AIP leaves the DF policy observation and the other RL rewards unchanged. Carry success then falls to zero, and sequential success falls to 5.2% at $N = 5$ and to zero for $N \geqslant 10$. The Push contact rate and the SitStand success at all three reported scales instead increase. This pattern suggests that matching interaction statistics is particularly useful for maintaining coordinated contact across multiple phases, whereas optimizing the remaining rewards can favor individual contact metrics. Motion-conditioned baselines likewise obtain higher scores in some settings where the geometry is compatible with their references (Tab. 2).

Discriminator input and numerical stability. The raw-DF representation ablation changes the policy observation but retains the encoded geometric input to AIP, and this configuration completes post-training. Replacing the latent input of the discriminator with raw DF features instead produced NaNs and prevented post-training from completing in our trials. These observations motivated retaining the latent interaction statistic in the reported representation comparison.

C.3 Vision Policy Observability Analysis

The vision policy replaces privileged distance-field observations with a depth-predicted interaction latent. This replacement makes deployment possible without MoCap but introduces partial observability. The limitation is most visible in SitStand, where the key pelvis–chair contact occurs behind the robot and is poorly observed by the front-facing egocentric camera.

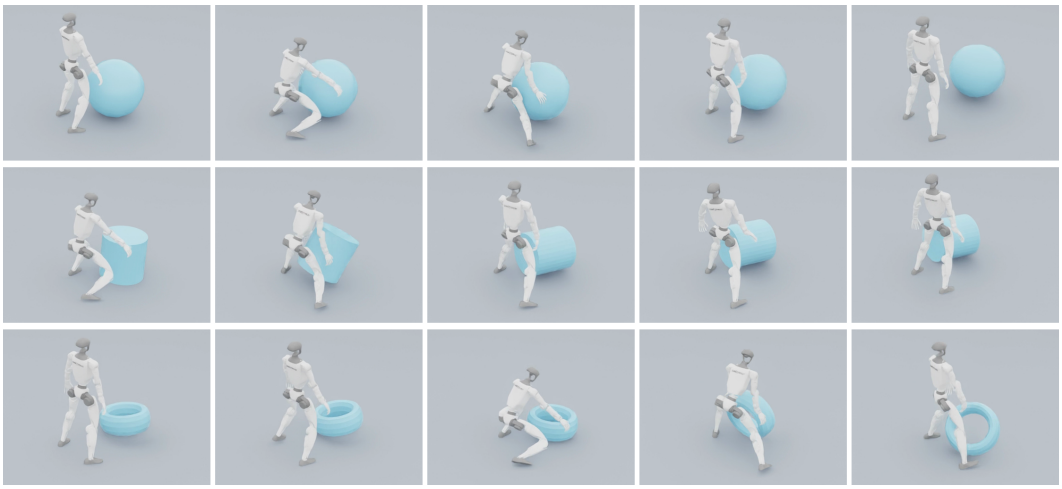


Figure A1: **Representative failures on curved objects.** Unsuccessful interactions with a sphere, a cylinder, and a torus.



Figure A2: **Gait of the mimic teacher and the post-trained policy.** Matched rollouts of π_{mimic} (top) and π_{full} (bottom) under the same configuration.

The front–rear depth ablation in Tab. A4 tests the role of visual coverage. The front-only model achieves 0% success at every scale, whereas adding a rear view raises the scale-averaged success to 31%. The remaining gap to privileged DF sensing may reflect residual occlusion and errors in estimating the interaction latent. The standard front-only hardware vision evaluation therefore covers PickUp, and quantitative SitStand trials use the MoCap-based policy.

The low sequential success of the vision policy is partly associated with the same observability limitation. When sequences involving SitStand are excluded, the vision policy reaches 36% success, which shows that the depth-predicted latent can support sequential execution when the relevant contact surfaces are visible. Improved camera placement, rear-facing sensors, or multi-view onboard perception could extend visual coverage to these interactions.

C.4 Failure Modes and Limitations

Most observed failures occur when an interaction requires timing, force precision, or visibility beyond what the current sensing and per-link geometric interface provide. Sensing latency in the MoCap or vision pipeline can delay contact during fast motions. Mismatched contact force can also cause failure, because excessive force may destabilize the object or the robot, whereas insufficient force can lead to missed grasps, weak pushing, or failed carrying. Objects that require precise force balance or fine contact placement remain challenging, since the per-link DF representation captures coarse proximity and surface direction rather than a detailed pressure distribution. For vision-based deployment, failures occur mainly when task-relevant contacts are occluded or outside the egocentric

camera view, such as the back-side chair contact in SitStand. Figure A1 shows representative failures on objects with curved surfaces.

Hardware motion also shows small steps, low foot clearance, and occasional jitter. Slight out-toeing and low-clearance gait patterns are already present in the retargeted references and in the mimic teacher, and post-training adds a modest degradation in gait quality. The interaction and whole-body style objectives may produce competing corrections under randomized geometry. Figure A2 compares matched rollouts of the mimic teacher and the post-trained policy. Sensing errors and latency further affect motion on hardware. Temporary errors in the MoCap root pose occur when the foam roll occludes the cameras, and the Python deployment stack adds roughly 5 to 15 ms of variable control latency.

Future improvements include reward balancing and temporal regularization for smoother motion, an optimized C++/ONNX control stack, force-aware feedback, denser geometric queries, and wider visual coverage.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims made in the abstract and introduction accurately reflect the paper's contributions.

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Sec. C.4 for the limitations.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper does not include theoretical results.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We have included necessary details in the paper and the supplementary material. The training code, data, checkpoint, and deployment system will be released on our project website.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We have included necessary details in the paper and the supplementary material. The training code, data, checkpoint, and deployment system will be released on our project website.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Yes. We have included these details in the paper.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Yes, we report them in our experiment section Sec. 4 and Sec. B.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We have included these details in Sec. B.2.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.

- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: The research conducted in the paper conforms with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: There is no societal impact of the work.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The assets are properly credited and the license are mentioned and respected.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not introduce or release a new public dataset, benchmark, or model asset. The supplementary material only provides implementation details, experimental protocols, and qualitative results needed to support reproducibility.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing or human-subjects experiments. Human operators are only used to provide high-level robot commands, such as joystick root commands and interaction-type flags, during system deployment and evaluation; the study does not collect or analyze human-subject data.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing or human-subjects experiments.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs were not used as a component of the core methodology or experimental design. Their use was limited to basic text refinement, such as spell checking and grammar suggestions.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.