ExtremControl: Low-Latency Humanoid Teleoperation with Direct Extremity Control

Ziyan Xiong*[†] Lixing Fang*[†] Junyun Huang*[†] Kashu Yamazaki[‡] Hao Zhang[†] Chuang Gan[†§]
[†]UMass Amherst [‡]Carnegie Mellon University [§]MIT-IBM Watson AI Lab *Equal Contributions

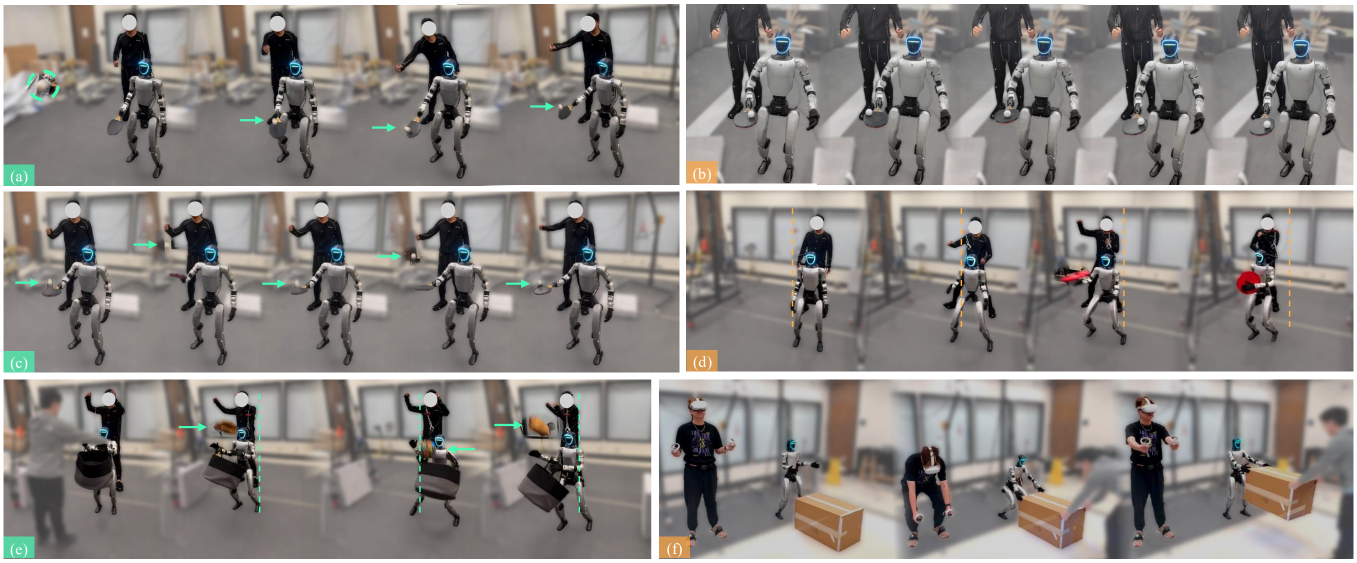


Fig. 1: The humanoid robot (Unitree G1) demonstrates a diverse set of loco-manipulation tasks under teleoperation: (a) returning a ping-pong ball from varying positions; (b) balancing a ping-pong ball on a paddle through rapid orientation adjustments; (c) juggling a ping-pong ball; (d) catching a frisbee while moving; (e) catching thrown objects using a handheld basket while in motion; and (f) cooperatively lifting a box. Tasks (a-e) are performed using an optical MoCap system to achieve lower latency, while task (f) is operated using a VR system.

***Abstract*—Building a low-latency humanoid teleoperation system is essential for collecting diverse reactive and dynamic demonstrations. However, existing approaches rely on heavily pre-processed human-to-humanoid motion retargeting and position-only PD control, resulting in substantial latency that severely limits responsiveness and prevents tasks requiring rapid feedback and fast reactions. To address this problem, we propose *ExtremControl*, a low latency whole-body control framework that: (1) operates directly on $\mathrm{SE}(3)$ poses of selected rigid links, primarily humanoid extremities, to avoid full-body retargeting; (2) utilizes a Cartesian-space mapping to directly convert human motion to humanoid link targets; and (3) incorporates velocity feedforward control at low level to support highly responsive behavior under rapidly changing control interfaces. We further provide a unified theoretical formulation of *ExtremControl* and systematically validate its effectiveness through experiments in both simulation and real-world environments. Building on *ExtremControl*, we implement a low-latency humanoid teleoperation system that supports both optical motion capture and VR-based motion tracking, achieving end-to-end latency as low as 50 ms and enabling highly responsive behaviors such as ping-pong ball balancing, juggling, and real-time return, thereby substantially surpassing the 200 ms latency limit observed in prior work.**

I. Introduction

Humanoid robots have long attracted significant attention in the robotics community due to their human-like morphology and kinematic structure. Because modern environments, tools, and tasks are predominantly designed around human bodies, humanoids represent a natural embodiment for general-purpose robotic systems capable of operating in unstructured, human-centric settings. Moreover, the close correspondence between humanoid and human morphology enables the direct exploitation of large-scale human motion and skill datasets, alleviating the reliance on expensive and limited robot-collected data. However, these same properties that make humanoids appealing also pose substantial challenges for traditional control frameworks. In particular, the high-dimensional state and action spaces, underactuated floating-base dynamics, intermittent contacts, and frequent hybrid mode transitions inherent to humanoid locomotion and manipulation render classical model-based control approaches difficult to scale and deploy robustly in practice. Accurate modeling, real-time optimization, and contact-consistent planning become increasingly intractable as task complexity grows, motivating the exploration of alternative control paradigms better suited to the complexity of humanoid systems.

With the advent of large-scale parallel simulation, reinforcement learning has become a dominant paradigm for humanoid locomotion and whole-body control, and the design of control interfaces has undergone a clear evolution over time. Inheriting from quadruped locomotion, early approaches relied on

Teleoperation System	Control Interface	Wrist Control	Foot Control	Full-Body Tracking	Joint-Space Retarget	End-to-End Latency
HOMIE [4]	Exoskeleton	✓	×	×	×	∼ 454 ms
HumanPlus [13]	RGB Camera	✓	✓	✓	✓	∼ 340 ms
OmniH2O [17]	VR	✓	×	×	×	**∼ 185 ms**
H2O [18]	RGB Camera	×	✓	✓	✓	∼ 373 ms
AMO [27]	VR	✓	×	×	✓	∼ 380 ms
CLONE [30]	VR	✓	×	×	×	**∼ 178 ms**
AMS [41]	MoCap	×	✓	✓	✓	∼ 201 ms
TWIST [56]	MoCap	×	✓	✓	✓	> 700 ms
TWIST2 [57]	VR	✓	✓	✓	✓	∼ 234 ms
ExtremControl (Ours)	VR, MoCap	✓	✓	✓	×	**∼ 54 ms**

TABLE I: Existing humanoid teleoperation systems. End-to-end latencies are estimated using optical flow on the authors' released videos.

explicit Cartesian-space objectives [17, 18] or command-based interfaces [58, 61]. Subsequently, influenced by advances in physics-based character animation [33, 42], many methods shifted toward dense joint-space pose supervision adopted from the human model, enabling humanoids to reproduce expressive and highly dynamic motions derived from human demonstrations [13, 20]. More recent efforts further narrowed the control objective to optimization of selected reference trajectories, achieving high-fidelity motion reproduction at the cost of generalization across tasks and behaviors [1, 6, 19, 31]. In parallel, teleoperation-oriented methods introduce intermediate representations to flexibly map human motion to humanoid embodiments [27, 41, 56, 57]. These approaches first compute target poses for a set of robot links and then solve inverse kinematics to obtain joint configurations that realize those poses. However, although policies are executed in joint space, the underlying optimization objective remains defined in terms of matching Cartesian link poses.

A large portion of the literature on humanoid control focuses on teleoperation [4, 13, 17, 18, 27, 30, 41, 56, 57], as it provides an effective mechanism for collecting data to train general-purpose robotic intelligence. Because teleoperation operates in a human-in-the-loop closed-loop setting, system latency critically determines the operator's ability to perform responsive tasks. Among the existing teleoperation systems listed in Tab. I, we observe a striking consistency: **most real-time humanoid teleoperation systems exhibit end-to-end latencies around 200 ms**, largely independent of the robot, retargeting strategy, or the length of future motion used [17, 30, 41, 56, 57].[1] We substantially surpass this apparent latency barrier by moving beyond the widely used position-only PD control paradigm. Instead, we introduce a velocity feedforward term that reduces the low-level control response time by approximately 100 ms, rendering the latency introduced by full-body human-to-humanoid retargeting (∼10 ms) non-negligible. Among prior methods that rely on position-only PD control, He et al. [17] and Li et al. [30] achieve the lowest latency by directly controlling the Cartesian positions of a selected set of robot links; however, foot links are excluded, which limits their capability to support complete whole-body control.

To address this problem, we introduce *ExtremControl*, a humanoid whole-body control framework that: 1) maps the human motion to $\mathrm{SE}(3)$ of selected rigid links, including all humanoid extremities, through a Cartesian-space mapping;
2) takes target link poses directly as policy input to avoid latency caused by joint-space retargeting; and 3) incorporates velocity feedforward control for extremely responsive low-level actuation, supported by a whole-body impedance calibration that tightly couples simulation with real-world deployment. The overall design is aimed at minimizing system latency while preserving full whole-body control capability.

The contributions of this work are fourfold. First, we present a unified theoretical formulation for *ExtremControl* from robot kinematics, dynamics, and policy learning. Second, we validate the effectiveness and optimality of this formulation through extensive experiments conducted in both simulation and on real humanoid platforms. Third, we introduce an optical-flow–based latency estimation method for measuring the end-to-end delay of teleoperation systems. Finally, leveraging *ExtremControl*, **we develop a humanoid teleoperation system that achieves as low as 50 ms latency**, thereby unlocking hardware capabilities that have remained dormant due to non-extreme system design. The system operates at near-perceptual latency, enabling fluid human-in-the-loop manipulation, and significantly enhancing both the capability and efficiency of teleoperation-based data collection.

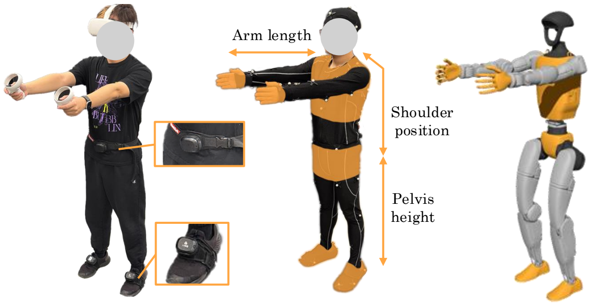
II. Kinematics

In this work, we deliberately avoid full-body human-to-humanoid retargeting within the policy loop. Instead, we formulate the control interface in terms of rigid link poses of selected robot links, which are obtained through Cartesian-space mapping from the corresponding human link poses.

A. Tracking Objectives

The choice of tracking objectives fundamentally determines the feasibility and robustness of a humanoid teleoperation system. Thus, selecting an appropriate subset of links as control interface is critical. It must be expressive enough to convey the operator's intent for manipulation, locomotion, and global posture, while remaining minimal to avoid over-constraining the system and amplifying sensing noise.

[1] We estimate latency by running optical flow on reported videos. Fu et al. [13] and He et al. [17, 18] used Unitree H1.



(a) VR (b) MoCap (c) Robot

Fig. 2: Tracking objectives for humans and humanoids under VR and MoCap settings.

a) Application Scenario: In most practical teleoperation scenarios, humanoid robots are not required to perform contact-rich whole-body manipulation. While such extreme cases exist [52], they are relatively rare and further complicated by scale mismatch between human and robot embodiments. Under this contact-sparse assumption, we focus on the most common and practically relevant configuration: hands for manipulation, feet for locomotion, and the torso and pelvis for representing the global body pose. Accordingly, as illustrated in Fig. 2, we select the highlighted links of Unitree G1, yielding $\mathbf{T}^r = [\mathbf{R}^r, \mathbf{p}^r]^6 \in \mathrm{SE}(3)^6$ where r stands for robot.

b) Kinematic Sufficiency: The six selected links are sufficient to describe meaningful whole-body motion while intentionally leaving internal joint configurations under-constrained. The Unitree G1 has seven degrees of freedom per arm and six per leg. Even though the Cartesian pose of hand does not uniquely determine the corresponding joint configuration, joint limits, self-collision constraints, and temporal continuity strongly restrict the feasible solution space, making smooth trajectories effectively identifiable from the tracked links. We further demonstrate in Sec. VI-C that directly using extremity poses achieves performance comparable to that obtained when retargeted joint configurations are included.

c) Teleoperation Input Modalities: Optical motion capture is restricted to limited capture volumes, video-based pose estimators yield noisy results, standard VR headsets do not provide foot tracking, and exoskeleton systems are inconvenient and expensive. In contrast, VR systems augmented with motion trackers (i.e. PICO 4 Ultra and VIVE Ultimate Tracker) provide six tracked poses, consisting of the headset, two hand controllers, and three trackers mounted on the waist and both feet. This configuration naturally aligns with the selected tracking objectives of Unitree G1, providing corresponding human link poses $\mathbf{T}^h \in \mathrm{SE}(3)^6$ where h stands for human.

B. Cartesian Space Mapping

Based on tracking links selected in Sec. II-A, we define a Cartesian-space mapping operator $\mathcal{M} : \mathrm{SE}(3)^6 \to \mathrm{SE}(3)^6$, which maps the set of 6 tracked human link poses $\mathbf{T}^h$ to the corresponding humanoid link targets $\mathbf{T}^r$. The design of $\mathcal{M}$ follows three principles: (i) anthropomorphic compensation: explicitly handling differences in body proportions; (ii) mathematical consistency: reproducing identical poses at calibration; and (iii) real-time efficiency: avoiding joint-space optimization. As a result, the proposed mapping produces smooth, scale-consistent $\mathrm{SE}(3)$ targets for a minimal set of humanoid links, incurs negligible computational overhead.

1) One-shot Calibration: The mapping is parameterized through a one-time calibration procedure performed at system initialization. During calibration, the human stands in a neutral pose as shown in Fig. 2. To resolve coordinate-system differences, we estimate a per-link rotational offset during calibration. These offsets are held fixed and are applied identically at run-time.

To compensate for body shape differences, we explicitly measure a small set of anthropomorphic measurements from $\mathbf{T}^h$, which is essential for respecting the relative positioning and motion limits of humanoid extremities:

- Pelvis height: $z^h_{\text{pelvis}} = \mathbf{p}^h_{\text{pelvis},z}$;
- Shoulder position in pelvis frame: $\mathbf{p}^h_{k\_\text{shoulder}} = \mathbf{p}^h_{k\_\text{hand},yz} - z^h_{\text{pelvis}}$;
- Arm length: $l^h_{k\_\text{arm}} = \mathbf{p}^h_{k\_\text{hand},x}$.

for each side $k \in \{\text{left}, \text{right}\}$.

The corresponding humanoid parameters $\{z^r_{\text{pelvis}}, \mathbf{p}^r_{k\_\text{shoulder}}, l^r_{k\_\text{arm}}\}$ are obtained directly from the humanoid kinematic model as constants.

Finally, we record a fixed foot offset $\delta_{k\_\text{foot}} = \mathbf{p}^r_{k\_\text{foot}} - (z^r_{\text{pelvis}}/z^h_{\text{pelvis}}) \cdot \mathbf{p}^h_{k\_\text{foot}}$ to ensure stable standing and pose uniformity. $\mathbf{p}^r_{k\_\text{foot}}$ represents the foot position in the humanoid frame when standing at the origin.

2) Per-frame Mapping: At run-time, given a new frame of tracked human poses $\mathbf{T}^h$, the humanoid target poses are computed as $\mathbf{T}^r = \mathcal{M}(\mathbf{T}^h)$:

Pelvis and feet are scaled according to the ratio between humanoid and human pelvis heights

$$\mathbf{p}^r_{\text{pelvis},k\_\text{foot}} = \frac{z^r_{\text{pelvis}}}{z^h_{\text{pelvis}}} \cdot \mathbf{p}^h_{\text{pelvis},k\_\text{foot}}. \tag{1}$$

Torso is not taken directly from the captured data. Its position is rigidly attached to the pelvis following the humanoid's kinematic structure:

$$\mathbf{p}^r_{\text{torso}} = \mathbf{p}^r_{\text{pelvis}} + \mathbf{R}^r_{\text{pelvis}} \mathbf{p}_{\text{diff}}, \tag{2}$$

where $\mathbf{p}_{\text{diff}}$ is the fixed pelvis-to-torso displacement in the humanoid model. The torso orientation is directly mapped.

Hands are mapped by first aligning the shoulder position in the pelvis frame and then scaling the arm length, such that $\mathbf{T}^r_{k\_\text{hand}}$ satisfies

$$\begin{aligned} &(\mathbf{T}^r_{k\_\text{hand}}(\mathbf{T}^r_{\text{torso}})^{-1} - \mathbf{p}^r_{k\_\text{shoulder}})/l^r_{k\_\text{arm}} \\ = \; &(\mathbf{T}^h_{k\_\text{hand}}(\mathbf{T}^h_{\text{torso}})^{-1} - \mathbf{p}^h_{k\_\text{shoulder}})/l^h_{k\_\text{arm}}. \end{aligned} \tag{3}$$

This computation consists solely of rigid transformations and can be directly calculated in closed form. Unlike joint-space retargeting, which is inherently sequential and must wait for the previous frame, our mapping is fully feedforward and parallelizable. Although this difference may be negligible at 50 Hz, retargeting will eventually fail to keep up as the control frequency increases.

III. Dynamics

Position-only PD control is widely adopted in learning-based locomotion methods, including all systems listed in Tab. I. However, the absence of a velocity term inevitably introduces substantial tracking delay, even under constant joint velocities. To overcome this limitation, we systematically formulate a **velocity feedforward control framework**.

A. Whole-Body Impedance Calibration

To facilitate a principled discussion of PD controller design, we calibrate joint-level PD gains using a sequential, simulation-based procedure that estimates effective joint impedance under whole-body closed-loop control. Joints are initialized with random gains and processed from distal to proximal to reduce coupling effects. For each joint in calibration, the derivative gain is temporarily set to zero and a small perturbation Δq is applied, inducing oscillations that are locally modeled by

$$M_{\text{eff}}\ddot{q} + k_p(q - q_0) = 0, \tag{4}$$

where M_{eff} captures both physical inertia and closed-loop coupling. To robustly estimate M_{eff}, we sample proportional gains $k_p^{(e)} \sim \mathcal{U}(0.5k_p, 1.5k_p)$ in different parallel simulation environments $e \in E$ and measure the corresponding oscillation periods $P^{(e)}$ in parallel simulations, yielding

$$M_{\text{eff}}^{(e)} = \frac{k_p^{(e)}\left(P^{(e)}\right)^2}{(2\pi)^2}. \tag{5}$$

The average $\overline{M_{\text{eff}}}$ is then used to update the PD gains with target natural frequency ω_n and damping ratio ζ as

$$k_p := \overline{M_{\text{eff}}}\,\omega_n^2, \qquad k_d := 2\zeta\overline{M_{\text{eff}}}\,\omega_n \tag{6}$$

This process is iterated over all joints to obtain dynamically consistent gains that provide a stable and well-scaled initialization for downstream control and learning.

B. Velocity Feedforward Control

We consider position-only PD control commonly adopted in learning-based locomotion,

$$\tau = k_p(q_t - q) - k_d\dot{q}, \tag{7}$$

where τ is the applied joint torque and q_t denotes the target joint position. We extend this formulation by incorporating a velocity feedforward term,

$$\tau = k_p(q_t - q) - k_d\dot{q} + \eta k_d\dot{q}_t, \tag{8}$$

where $\eta \in [0, 1]$ is the velocity feedforward ratio.

For analysis, we model the joint dynamics as $M_{\text{eff}}\ddot{q} = \tau$. Following the whole-body impedance calibration described in III-A, we parameterize the PD gains using a target natural frequency ω_n with critical damping $\zeta = 1$. Substituting into the closed-loop dynamics yields

$$\ddot{q} + 2\omega_n\dot{q} + \omega_n^2 q = 2\eta\,\omega_n\dot{q}_t + \omega_n^2 q_t. \tag{9}$$

Since the closed-loop system is linear and time-invariant, its response to an arbitrary reference trajectory can be expressed as a superposition of responses to sinusoidal components via Fourier decomposition. Therefore, without loss of generality, we analyze a sinusoidal reference

$$q_t(t) = A\sin(\omega t), \tag{10}$$

which fully characterizes the frequency-dependent tracking behavior of the controller.

Taking the Laplace transform, the transfer function from the reference q_t to the realized joint position q is

$$H(s) = \frac{\omega_n^2 + 2\eta\,\omega_n s}{s^2 + 2\omega_n s + \omega_n^2}. \tag{11}$$

For sinusoidal input $q_t(t)$, the steady-state response takes the form

$$q(t) = A|H(j\omega)|\sin\big(\omega t + \phi(\omega)\big), \tag{12}$$

where the phase difference between the realized motion and the reference is given by

$$\phi(\omega) = \tan^{-1}\left(2\eta\frac{\omega}{\omega_n}\right) - \tan^{-1}\left(\frac{2(\omega/\omega_n)}{1 - (\omega/\omega_n)^2}\right). \tag{13}$$

In the low-frequency regime $\omega \ll \omega_n$, the phase shift can be interpreted as an equivalent tracking delay,

$$\ell = -\frac{\phi(\omega)}{\omega} \approx \frac{2(1-\eta)}{\omega_n}. \tag{14}$$

C. Discrete-Time Compensation

In practice, policy inference and communication impose a fixed control period Δt, during which the target command is held constant. We analyze the effect of velocity feedforward under this discrete-time execution using a conservative local approximation.

We assume that at the beginning of a control interval the joint matches the reference ($q \approx q_t$), and that the desired trajectory evolves linearly within the interval with constant velocity $\dot{q}_t$. Under zero-order hold, the average deviation between the true reference and the held target over one control step is

$$\overline{\Delta q_t} \approx \frac{\dot{q}_t\Delta t}{2}. \tag{15}$$

The corresponding proportional and feedforward torque contributions are

$$\tau_p \approx k_p\frac{\dot{q}_t\Delta t}{2}, \qquad \tau_v = \eta k_d\dot{q}_t. \tag{16}$$

To avoid additional acceleration within a single control interval that may lead to overshoot, we require

$$\tau_p + \tau_v \leq k_d\dot{q}_t \tag{17}$$

which yields an upper bound on the feedforward ratio,

$$\eta \leq 1 - \frac{\omega_n\Delta t}{4}. \tag{18}$$

IV. Policy Learning

Existing human motion datasets are large in scale and diverse in difficulty. To ensure reproducibility and facilitate systematic parameter optimization, we adopt a three-stage learning framework. First, we train a teacher policy π_{teacher} using fine-grained, high-difficulty motion data together with privileged observations, enabling robust handling of dynamic motions. Second, we distill a deployable student policy π_{student} that operates with a single future motion frame. Finally, we fine-tune the distilled policy on a large and diverse motion dataset to obtain the teleoperation policy π_{teleop}.

A. Observation

Since our control interface operates solely on a selected set of robot link poses, we introduce a future interpretation function to encode reference motion over a finite horizon H

$$\mathbf{o}^{\text{ref}}(H) = \mathscr{I}(\mathbf{T}^r_{t:t+H}, \mathbf{V}^r_{t:t+H}, \mathbf{T}^r_{t-1}) \tag{19}$$

where $\mathbf{V}^r = \dot{\mathbf{T}}^r(\mathbf{T}^r)^{-1}$ represents the target velocities in the global frame. The detail of $\mathscr{I}$ is provided in the appendix. To explicitly capture extremity pose errors and account for the IMU being mounted on the pelvis of the Unitree G1, we express both target poses $\mathbf{T}^{r\prime}$ and actual poses $\mathbf{U}^{r\prime}$ in the pelvis frame, where $\mathbf{U}^{r\prime}$ is obtained via forward kinematics from the pelvis, and compute the local pose discrepancy as

$$\mathbf{o}^{\text{diff}} = (\mathbf{U}^{r\prime})^{-1}\mathbf{T}^{r\prime}. \tag{20}$$

For deployable policies π_{student} and π_{teleop}, we use a single future frame to maximize responsiveness, resulting in the observation vector $[\mathbf{o}^{\text{ref}}(1), \mathbf{o}^{\text{diff}}, \mathbf{o}^{\text{proprio}}]$ where $\mathbf{o}^{\text{proprio}}$ denotes proprioceptive observations. The oracle policy π_{teacher} leverages a longer horizon together with privileged observations $\mathbf{o}^{\text{priv}}$, including global-frame pose discrepancies, joint configurations after inverse kinematics, domain randomization parameters and foot contact forces, yielding $[\mathbf{o}^{\text{ref}}(32), \mathbf{o}^{\text{diff}}, \mathbf{o}^{\text{proprio}}, \mathbf{o}^{\text{priv}}]$. The complete specification of observations is provided in the appendix.

B. Reward Function

The reward function is composed of two categories: tracking rewards and regularization rewards. In addition to penalizing discrepancies between the target and actual poses of the selected links, we introduce an auxiliary reward that tracks retargeted joint configurations. This term encourages exploration during training, including for the teleoperation policy π_{teleop}, which does not directly observe the retargeted joint configuration. We adopt GMR [2] for full-body joint-space retargeting with optimization over a small subset of retargeting targets. The regularization terms are primarily designed to constrain joint torques and suppress torso oscillations.

To improve exploration stability in online reinforcement learning, we formulate all reward terms as negative-valued and compute the exponential of their weighted sum as the total reward. This formulation ensures that (i) the per-step total reward lies within $(0, 1)$, and (ii) the gradient of the total reward with respect to each individual term is preserved. The complete specification of the reward functions is provided in the appendix.

C. Policy Learning

In our setting, retargeted joint configurations are not included in the observation space of π_{teleop}, which makes exploration of complex motions from scratch particularly challenging. Compared to pure reinforcement learning (RL) [13] or reinforcement learning followed by behavior cloning (RL+BC) [56], we find that an RL+BC+RL training paradigm offers greater flexibility across learning stages and improved exploration stability. Specifically, we decompose the learning of a general tracking policy into three stages: (1) training an oracle teacher policy that handles highly dynamic and difficult motions using privileged observations and longer future horizons; (2) distilling the teacher into a deployable student policy by removing observations unavailable in real-world deployment and reducing the future horizon to minimize teleoperation latency; and (3) expanding the distilled policy over a broad motion dataset to obtain a task-agnostic whole-body controller.

We employ PPO [47] for online training of both π_{teacher} and π_{teleop}, using an entropy curriculum that gradually anneals the entropy coefficient to zero to encourage full exploitation at the end of training. During the second RL stage, to prevent instability caused by an untrained critic overwriting the distilled policy, we freeze the actor network for the first 200 training iterations, allowing the critic to converge under a fixed policy. Behavior cloning in the distillation stage is performed using DAgger [44], while the critic network is not trained at this stage, as PPO relies on stochastic action sampling whereas DAgger employs deterministic expert actions, leading to inconsistent value targets.

V. Experiment Setup

A. Simulation Setup

We leverage Genesis [3] as the simulation backbone, with a simulation timestep of $\text{sim\_dt} = \frac{1}{200}\,\text{s}$ and a single physics substep across all experiments, including whole-body impedance calibration, velocity feedforward ratio validation, and policy training. During policy training, we run 8,192 parallel environments, achieving over 100k policy steps per second on an NVIDIA L40s GPU with a control decimation of 4. Detailed simulation parameters and domain randomizations are provided in the appendix.

B. Real-World Robot Setup

Following common practice in control literature [16], we set the PD gains in simulation based on a target natural frequency of $\omega_n = 10\,\text{rad/s}$ with a damping ratio $\zeta = 1$. The velocity feedforward ratio is set to 0.9, while it is disabled for all indirect-drive joints, which are ankle and waist joints for Unitree G1. The low-level PD controller operates at $1000\,\text{Hz}$, whereas the high-level control policy runs at $50\,\text{Hz}$. This setting yields an upper bound on the velocity feedforward ratio of 0.95. To ensure consistent forward kinematics in observation computation, we apply a low-pass filter with coefficient $\alpha = 0.1$ at 1000 Hz to smooth the measured joint configuration.

C. Motion Dataset

As observed in prior work [56, 57], incorporating in-domain motion data significantly improves policy performance during teleoperation. We collect a set of motion sequences $\mathcal{S}_{\text{teleop}}$ using an optical motion capture system; however, to ensure fair evaluation and reproducibility, these user-specific datasets are excluded from policy training experiments unless explicitly stated.

Aiming for task-agnostic policy learning, we use the widely adopted LAFAN1 dataset $\mathcal{S}_{\text{lafan}}$ [15] retargeted by Unitree [35], as dynamic and challenging motions, and AMASS $\mathcal{S}_{\text{amass}}$ [36] as a large-scale and diverse motion corpus. Unless otherwise specified, we use $\mathcal{S}_{\text{lafan}}$ in teacher and student learning stages, $\mathcal{S}_{\text{lafan}} \cup \mathcal{S}_{\text{amass}}$ in RL finetune stage. For evaluation, we additionally collect a trajectory $\mathcal{S}'_{\text{teleop}} = \{s_{\text{eval}}\}, s_{\text{eval}} \notin \mathcal{S}_{\text{teleop}}$ comprising common daily motions to assess teleoperation performance, and we use a subset of LAFAN1 $\mathcal{S}'_{\text{lafan}} \subset \mathcal{S}_{\text{lafan}}$ as the dynamic-motion evaluation benchmark.

D. MoCap System

We employ an optical motion capture system, OptiTrack [24], to obtain the global position and rotation of human body links. Although the OptiTrack Motive software is capable of reconstructing a full-body human pose, our system only utilizes the subset of links described in Sec. II-A, which are sufficient for the proposed Cartesian-space mapping and tracking framework.

Human motion is streamed at 120 Hz. In contrast to TWIST [56], we implement a non-blocking streaming pipeline specifically designed for real-time control rather than offline recording. The pipeline minimizes transmission latency such that, when link poses are queried, the effective delay is bounded within a single MoCap streaming cycle ($\frac{1}{120}$ s). In a local setup, the system latency is less than 10 ms.

E. VR System

We employ a VR-based input system composed of a Meta Quest 2 [39] headset, hand controllers, and three VIVE Ultimate Trackers [8], with SteamVR [9] used as a unifying middleware to resolve cross-device compatibility. The trackers are mounted on the waist and both feet as in Fig. 2, providing direct pose measurements for the corresponding body links.

Because the torso pose is not directly tracked, we estimate the torso orientation from the planar motion of the headset expressed in the pelvis frame. Specifically, we compute the rotation that aligns the vertical axis $\vec{z}$ with $(\mathbf{R}_{\text{pelvis}})^{-1}\mathbf{p}_{\text{headset}}$, along the rotation axis orthogonal to the plane spanned by them. The headset's own orientation is intentionally not used, allowing the user to freely observe the environment.

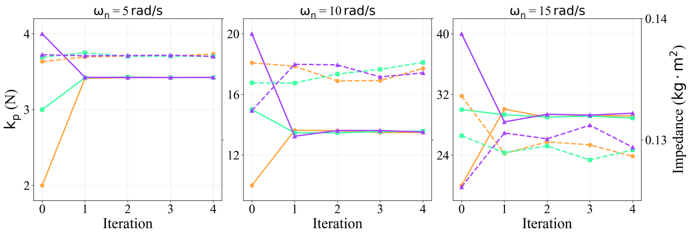


Fig. 3: Whole-body impedance calibration of Unitree G1 elbow joint. Solid lines correspond to proportional gains k_p, dashed lines depict the effective impedance.

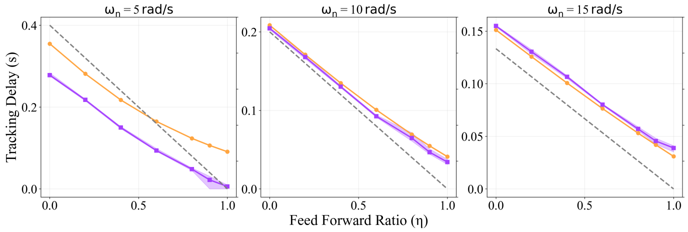


Fig. 4: Measured tracking delay in simulation and real world as a function of velocity feedforward ratio. Dashed line represents the theoretical value $\frac{2(1-\eta)}{\omega_n}$.

VI. Experiment Results

A. Whole-Body Impedance Calibration

As shown in Fig. 3, we evaluate the proposed whole-body impedance calibration under different target natural frequencies ω_n and initial proportional gains k_p on the Unitree G1 elbow joint, while three additional distal joints are calibrated simultaneously but not visualized. The derivative gain k_d is computed using a fixed damping ratio $\zeta = 1$ as defined in Eq. (6). Despite large variations in initialization, the calibration process consistently converges to stable effective impedance values that differ across ω_n, demonstrating robustness to initial gains and reliable recovery of M_{eff} induced by whole-body coupling.

B. Velocity Feedforward Control

Using the PD gains obtained from whole-body impedance calibration, we evaluate the low-level tracking delay predicted by Eq. (14) in both simulation and on hardware. A sinusoidal reference with frequency $\omega = 3.14\,\text{rad/s}$ is applied as the target joint position q_t, and the tracking delay is estimated via minimum cross-correlation between the target and measured joint positions.

As shown in Fig. 4, the theoretical predictions closely match the measured delays for $\omega_n = 10\,\text{rad/s}$ and $\omega_n = 15\,\text{rad/s}$, with the remaining discrepancy explained by the control period $\Delta t = 0.02, \text{s}$. For $\omega_n = 5\,\text{rad/s}$, deviations arise from violation of the assumption $\omega \ll \omega_n$ and numerical inaccuracies. With a control update frequency of 50 Hz, the measured delay increases at higher velocity feedforward ratios, producing the upward trend observed at the right end of the curves. This effect is consistent with the discrete-time overshoot described in Sec. III-C and reflects a bias in the delay estimate rather than a true increase in physical response latency; further analysis is provided in Sec. VI-D.

Tracking Error ↓	E_{pos}	$E_{\text{l\_pos}}$	$E_{\text{l\_rot}}$	E_{pos}	$E_{\text{l\_pos}}$	$E_{\text{l\_rot}}$
(a) Ablation on Velocity Feedforward Ratio η						
		$\pi_{\text{teleop}} \times \mathcal{S}'_{\text{lafan}}$			$\pi_{\text{teleop}} \times \mathcal{S}'_{\text{teleop}}$	
$\eta = 0.0$	0.34	0.103	0.42	**0.074**	**0.067**	0.24
$\eta = 0.2$	0.33	**0.101**	0.40	0.091	0.070	**0.22**
$\eta = 0.4$	0.34	0.103	0.41	0.085	0.067	0.25
$\eta = 0.6$	0.32	0.104	0.40	0.093	0.069	0.25
$\eta = 0.8$	**0.31**	0.103	**0.38**	0.075	**0.067**	**0.22**
$\eta = 0.9$ (Ours)	0.32	0.104	0.41	0.080	0.071	0.24
$\eta = 1.0$	0.36	**0.101**	0.41	0.113	0.075	0.26
(b) Ablation on Observed Future Length l						
		$\pi_{\text{teacher}} \times \mathcal{S}'_{\text{lafan}}$			$\pi_{\text{student}} \times \mathcal{S}'_{\text{lafan}}$	
$l = 1$	0.19	0.076	0.37	0.28	0.087	0.42
$l = 2$	0.19	0.077	0.37	0.28	0.087	0.41
$l = 4$	0.18	0.078	0.36	0.29	0.090	0.41
$l = 8$	0.17	0.076	**0.34**	0.28	0.087	**0.39**
$l = 16$	0.16	0.076	0.35	0.26	0.088	0.40
$l = 32$ (Ours)	**0.14**	0.074	**0.34**	**0.25**	**0.085**	**0.39**
$l = 64$	**0.14**	**0.072**	0.35	0.28	**0.085**	0.40
(c) Ablation on Learning Strategy						
		$\mathcal{S}'_{\text{lafan}}$			$\mathcal{S}'_{\text{teleop}}$	
RL	0.49	0.125	0.49	0.11	0.075	0.24
RL + BC	**0.25**	**0.085**	0.39	0.098	0.061	0.38
RL + BC [a]	0.33	0.110	0.48	0.095	0.067	0.27
Ours	0.30	0.099	0.41	0.080	0.064	0.24
Ours + $\mathcal{S}_{\text{teleop}}$ [b]	0.31	0.099	0.41	**0.076**	0.062	0.22
Ours + $\mathbf{o}^{q_t}$ [bc]	0.31	0.093	**0.38**	0.087	**0.055**	**0.20**

[a] Add $\mathcal{S}_{\text{amass}}$ in teacher and student learning stages.
[b] Add $\mathcal{S}_{\text{teleop}}$ in all learning stages.
[c] The retargeted joint configuration q_t is observable in student and teleoperation policy learning stages.

TABLE II: Ablations on Policy Learning. E_{pos} (m) represents the global tracking position error; $E_{\text{l\_pos}}$ (m) represents the links tracking position error in pelvis frame; $E_{\text{l\_rots}}$ (rad) represents the links tracking rotation error in pelvis frame.

C. Policy Learning Ablations

We evaluate the sensitivity of policy performance to the velocity feedforward ratio η. As shown in Tab. II (a), performance remains largely stable across a wide range of η, with global tracking error E_{pos} within a few centimeters and local link error $E_{\text{l\_pos}}$ below 1 cm. In contrast, $\eta = 1.0$ leads to severe overshooting and noticeable performance degradation, corroborating our theoretical prediction that η must be bounded under limited high-level control frequencies. This empirically validates the upper bound derived in Sec. V-B. Combined with the latency analysis in Tab. III, selecting $\eta \in \{0.8, 0.9\}$ represents a trade-off between tracking accuracy (4 mm, 0.02 rad) and latency (5 ms). Notably, as shown in Tab. III, attaching an additional 2 lb weight to each hand increases statistically detectable latency.

We further observe an intriguing phenomenon in the policy distillation process: because the student policy is constrained to operate with only a single future frame, a stronger teacher does not necessarily lead to a stronger student. In particular, as shown in Tab. II (b), we find that allowing the teacher to observe up to 32 future frames represents a practical sweet spot. Increasing the teacher's future horizon beyond this point leads to degraded performance after distillation. This suggests a fundamental capacity mismatch between the teacher and student policies, highlighting the importance of aligning the teacher's information horizon with the student's representational and observational constraints.

We ablate different learning strategies in Table II (c) and draw the following conclusions. (1) RL+BC trained on dynamic motions $\mathcal{S}_{\text{lafan}}$ achieves the best performance on dynamic tasks, whereas the proposed RL+BC+RL paradigm attains the best overall performance on the unseen trajectory set $\mathcal{S}'_{\text{teleop}}$ among the three learning strategies. (2) Consistent with prior observations [56, 57], incorporating in-domain data significantly improves tracking performance on $\mathcal{S}'_{\text{teleop}}$. (3) Eliminating joint-space retargeting introduces marginal tracking errors of 7 mm in position and 0.02 rad in orientation, which represent a favorable trade-off for the 10 ms latency reduction it enables.

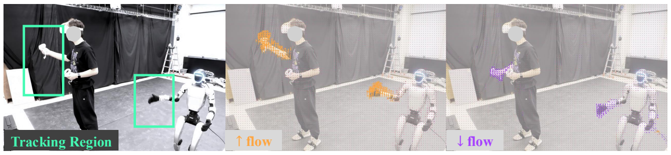


Fig. 5: Optical flow for latency analysis.

η	0.0	0.2	0.4	0.6	0.8	0.9	0.9[a]
ℓ_{overall} (ms)	155	131	104	82	69	64	65
ℓ_{control} (ms)	205	168	130	92	62	47	–

[a] We attach an additional 0.91 kg weight to each rubber hand.

TABLE III: Ablation of end-to-end latency under different velocity feedforward ratios, with the teleoperation system operating in VR mode.

D. Latency Analysis

We evaluate the end-to-end system latency using a video-based analysis, which does not rely on internal timestamps and provides **externally observable cumulative latency** across all systems. The latency is estimated directly from recorded videos by analyzing motion consistency between the human operator and the humanoid robot using optical flow. As shown in Fig. 5, we define tracking regions on both the human and robot that exhibit clear directional motion, then compute the optical flow [10] between consecutive frames and average it within each region. The resulting flow vectors are projected onto a predefined motion direction (e.g., vertical up–down in Fig. 5), producing a one-dimensional motion signal per frame.

To facilitate robust temporal alignment, we perform a simple reciprocating motion by hand during video recording, resulting in a quasi-periodic motion signal with clear phase structure. For other systems with video demonstrations, we select video clips and assign tracking areas that contain reciprocating motions or motions closely approximating this pattern, as shown in the middle column of Fig. 6, enabling fair comparison. We

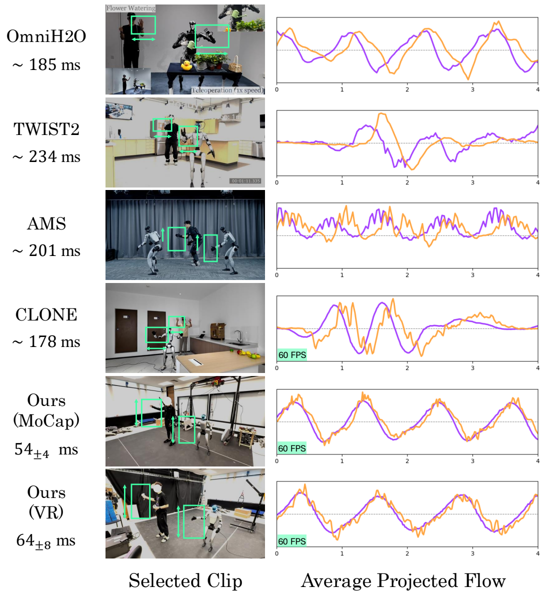


Fig. 6: Measured latencies from selected video clips. Normalized optical-flow projections of human and robot are displayed on the right side.

standardize the one-dimensional motion signal for both tracking areas. The system latency is estimated by measuring the temporal offset between the two sets of signals via waveform alignment, which is robust to amplitude differences.

We apply this analysis across multiple teleoperation systems, including prior teleoperation systems [17, 30, 41, 57] and our system using both optical motion capture and VR input. As shown in Fig. 6, our method exhibits consistently tighter phase alignment between human and robot motion, indicating lower end-to-end latency. Quantitatively, as shown in Fig. 6 and Tab. I, all the existing systems exhibit an overall latency exceeding 170 ms except for ours: the MoCap-based setup achieves an **average latency of 54 ± 4 ms**, while the VR-based setup achieves 64 ± 8 ms.

Furthermore, we analyze the end-to-end latency under different velocity feedforward ratios, as reported in Table III. A linear regression yields the following approximation:

$$\ell_{\text{overall}} = 0.58 \cdot \ell_{\text{control}} + 32\,\text{ms} \tag{21}$$

with $R^2 = 0.99$. This result indicates that low-level tracking delay does not translate one-to-one into end-to-end system latency, as evidenced by the ideal case of a policy that perfectly tracks a single trajectory, for which the overall latency would approach zero despite nonzero low-level delay due to the motion prior encoded in the policy. The remaining offset of 32 ms is attributed to communication from the VR system to the host PC, Cartesian-space mapping, policy inference, and the finite update rate of the control targets.

VII. Related Work

A. Reinforcement Learning for Locomotion

Reinforcement learning has become a dominant paradigm for learning agile locomotion policies, due to its ability to optimize high-dimensional control objectives directly from interaction. Early work demonstrated that model-free RL can produce robust locomotion behaviors in simulation and transfer them to real robots through domain randomization and privileged training signals [12, 26, 46, 51]. Subsequent studies improved robustness and versatility by introducing curriculum learning, command-conditioned policies [5, 7, 23, 38, 45, 60]. Hybrid approaches further combine trajectory optimization, model-based priors, or analytical controllers with RL fine-tuning to enhance stability and tracking accuracy [25, 29, 54, 55]. Together, these works establish RL as a practical framework for locomotion across diverse tasks and robot embodiments.

B. Humanoid Whole-Body Control

Compared to quadrupeds, humanoid locomotion and loco-manipulation involve high degrees of freedom, underactuated contacts, and complex whole-body coordination. Whole-body control (WBC) frameworks address these challenges by tracking joint-space or task-space objectives under full-body dynamics and contact constraints [7, 11, 21, 22, 48, 49]. Recent approaches integrate reinforcement learning with WBC, using motion capture and animation data to provide expressive reference motions [14, 28, 33, 34, 35, 37, 41, 42, 52, 59]. Humanoid teleoperation, which is a fundamental mechanism for large-scale data collection, fits naturally into this paradigm. Early methods specify Cartesian objectives for selected body links [4, 17, 30, 40, 43, 53] with control interfaces ranging from exoskeleton to single rgb camera; whereas recent approaches relied on full-body joint-space retargeting, incurring added latency [1, 2, 18, 32, 50, 56]. Overall, WBC serves as a unifying backbone for complex humanoid systems.

VIII. Conclusion

In this work, we present *ExtremControl*, a humanoid whole-body control framework designed to minimize teleoperation latency while preserving full whole-body control capability. Building on *ExtremControl*, we develop a humanoid teleoperation system that achieves end-to-end latencies as low as 50 ms and demonstrate its effectiveness on a range of highly responsive tasks. Despite these advances, several limitations remain. First, the Unitree G1 has seven DoFs in each arm, which introduces inverse kinematics ambiguity. When combined with direct extremity pose mapping and the evenly distribution of the wrist and elbow joints along the forearm, this can lead to unnatural arm poses. Second, while our work primarily targets on the responsiveness arising from control design, lower-body latency is governed by policy-level regulation of the center of mass, such as distinguishing whether the teleoperator intends to initiate walking or simply lift a foot. Third, our experiments use non-articulated rubber hand; extending the system to parallel grippers or dexterous hands introduces additional actuation and communication latencies. We aim to address these limitations in future work and move toward a near-human, low-latency humanoid data collection platform for general-purpose robotic intelligence.

REFERENCES

[1] Arthur Allshire, Hongsuk Choi, Junyi Zhang, David McAllister, Anthony Zhang, Chung Min Kim, Trevor Darrell, Pieter Abbeel, Jitendra Malik, and Angjoo Kanazawa. Visual imitation enables contextual humanoid control. In *Proceedings of the Conference on Robot Learning (CoRL)*, 2025.

[2] Joao Pedro Araujo, Yanjie Ze, Pei Xu, Jiajun Wu, and C. Karen Liu. Retargeting matters: General motion retargeting for humanoid motion tracking, 2025. URL https://arxiv.org/abs/2510.02252.

[3] Genesis Authors. Genesis: A generative and universal physics engine for robotics and beyond, December 2024. URL https://github.com/Genesis-Embodied-AI/Genesis.

[4] Qingwei Ben, Feiyu Jia, Jia Zeng, Junting Dong, Dahua Lin, and Jiangmiao Pang. HOMIE: Humanoid Loco-Manipulation with Isomorphic Exoskeleton Cockpit. In *Proceedings of Robotics: Science and Systems*, LosAngeles, CA, USA, June 2025. doi: 10.15607/RSS.2025.XXI.070.

[5] Penghui Chen, Yushi Wang, Changsheng Luo, Wenhan Cai, and Mingguo Zhao. Hifar: Multi-stage curriculum learning for high-dynamics humanoid fall recovery, 2025. URL https://arxiv.org/abs/2502.20061.

[6] Zixuan Chen, Mazeyu Ji, Xuxin Cheng, Xuanbin Peng, Xue Bin Peng, and Xiaolong Wang. Gmt: General motion tracking for humanoid whole-body control, 2025. URL https://arxiv.org/abs/2506.14770.

[7] Xuxin Cheng, Yandong Ji, Junming Chen, Ruihan Yang, Ge Yang, and Xiaolong Wang. Expressive Whole-Body Control for Humanoid Robots. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, July 2024. doi: 10.15607/RSS.2024.XX.107.

[8] HTC Corporation. Vive ultimate tracker - full-body tracking, steamvr support, 2026. URL https://www.vive.com/us/accessory/vive-ultimate-tracker/.

[9] Valve Corporation. Steamvr - valve corporation, 2026. URL https://www.steamvr.com/.

[10] Gunnar Farnebäck. Two-frame motion estimation based on polynomial expansion. In *Scandinavian conference on Image analysis*, pages 363–370. Springer, 2003.

[11] Siyuan Feng, Eric Whitman, X Xinjilefu, and Christopher G. Atkeson. Optimization based full body control for the atlas robot. In *2014 IEEE-RAS International Conference on Humanoid Robots*, pages 120–127, 2014. doi: 10.1109/HUMANOIDS.2014.7041347.

[12] Justin Fu, Katie Luo, and Sergey Levine. Learning robust rewards with adverserial inverse reinforcement learning. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=rkHywl-A-.

[13] Zipeng Fu, Qingqing Zhao, Qi Wu, Gordon Wetzstein, and Chelsea Finn. Humanplus: Humanoid shadowing and imitation from humans. In *Conference on Robot Learning (CoRL)*, 2024.

[14] Jinrui Han, Weiji Xie, Jiakun Zheng, Jiyuan Shi, Weinan Zhang, Ting Xiao, and Chenjia Bai. Kungfubot2: Learning versatile motion skills for humanoid whole-body control. *arXiv:2509.16638*, 2025.

[15] Félix G Harvey, Mike Yurick, Derek Nowrouzezahrai, and Christopher Pal. Robust motion in-betweening. *ACM Transactions on Graphics (TOG)*, 39(4):60–1, 2020.

[16] Galal A. Hassaan. Tuning of a pd controller used with second order processes. *International Journal of Engineering*, 2, 2014. URL https://api.semanticscholar.org/CorpusID:195955908.

[17] Tairan He, Zhengyi Luo, Xialin He, Wenli Xiao, Chong Zhang, Weinan Zhang, Kris Kitani, Changliu Liu, and Guanya Shi. Omnih2o: Universal and dexterous human-to-humanoid whole-body teleoperation and learning. *arXiv preprint arXiv:2406.08858*, 2024.

[18] Tairan He, Zhengyi Luo, Wenli Xiao, Chong Zhang, Kris Kitani, Changliu Liu, and Guanya Shi. Learning human-to-humanoid real-time whole-body teleoperation. *arXiv preprint arXiv:2403.04436*, 2024.

[19] Tairan He, Jiawei Gao, Wenli Xiao, Yuanhang Zhang, Zi Wang, Jiashun Wang, Zhengyi Luo, Guanqi He, Nikhil Sobanbabu, Chaoyi Pan, Zeji Yi, Guannan Qu, Kris Kitani, Jessica K. Hodgins, Linxi Fan, Yuke Zhu, Changliu Liu, and Guanya Shi. ASAP: Aligning Simulation and Real-World Physics for Learning Agile Humanoid Whole-Body Skills. In *Proceedings of Robotics: Science and Systems*, LosAngeles, CA, USA, June 2025. doi: 10.15607/RSS.2025.XXI.066.

[20] Tairan He, Wenli Xiao, Toru Lin, Zhengyi Luo, Zhenjia Xu, Zhenyu Jiang, Jan Kautz, Changliu Liu, Guanya Shi, Xiaolong Wang, Linxi Jim Fan, and Yuke Zhu. Hover: Versatile neural whole-body controller for humanoid robots. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9989–9996, 2025. doi: 10.1109/ICRA55743.2025.11128549.

[21] Bernd Henze, Máximo A. Roa, and Christian Ott. Passivity-based whole-body balancing for torque-controlled humanoid robots in multi-contact scenarios. *The International Journal of Robotics Research*, 35(12): 1522–1543, 2016. doi: 10.1177/0278364916653815. URL https://doi.org/10.1177/0278364916653815.

[22] Alexander Herzog, Ludovic Righetti, Felix Grimminger, Peter Pastor, and Stefan Schaal. Balancing experiments on a torque-controlled humanoid with hierarchical inverse dynamics. *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 981–988, 2013. URL https://api.semanticscholar.org/CorpusID:3104000.

[23] Jemin Hwangbo, Joonho Lee, Alexey Dosovitskiy, Dario Bellicoso, Vassilios Tsounis, Vladlen Koltun, and Marco Hutter. Learning agile and dynamic motor skills for legged robots. *Science Robotics*, 4(26):eaau5872, 2019. doi: 10.1126/scirobotics.aau5872. URL https://www.science.org/doi/abs/10.1126/scirobotics.aau5872.

[24] NaturalPoint Inc. Optitrack - motion capture systems,

2026. URL https://www.optitrack.com/.

[25] Tobias Johannink, Shikhar Bahl, Ashvin Nair, Jianlan Luo, Avinash Kumar, Matthias Loskyll, Juan Aparicio Ojea, Eugen Solowjow, and Sergey Levine. Residual reinforcement learning for robot control. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 6023–6029, 2019. doi: 10.1109/ICRA.2019.8794127.

[26] Ashish Kumar, Zipeng Fu, Deepak Pathak, and Jitendra Malik. Rma: Rapid motor adaptation for legged robots. 2021.

[27] Jialong Li, Xuxin Cheng, Tianshu Huang, Shiqi Yang, Ri-Zhao Qiu, and Xiaolong Wang. AMO: Adaptive Motion Optimization for Hyper-Dexterous Humanoid Whole-Body Control. In *Proceedings of Robotics: Science and Systems*, LosAngeles, CA, USA, June 2025. doi: 10.15607/RSS.2025.XXI.061.

[28] Yitang Li, Zhengyi Luo, Tonghe Zhang, Cunxi Dai, Anssi Kanervisto, Andrea Tirinzoni, Haoyang Weng, Kris Kitani, Mateusz Guzek, Ahmed Touati, Alessandro Lazaric, Matteo Pirotta, and Guanya Shi. Bfm-zero: A promptable behavioral foundation model for humanoid control using unsupervised reinforcement learning, 2025. URL https://arxiv.org/abs/2511.04131.

[29] Yitang Li, Yuanhang Zhang, Wenli Xiao, Chaoyi Pan, Haoyang Weng, Guanqi He, Tairan He, and Guanya Shi. Hold my beer: Learning gentle humanoid locomotion and end-effector stabilization control, 2025. URL https://arxiv.org/abs/2505.24198.

[30] Yixuan Li, Yutang Lin, Jieming Cui, Tengyu Liu, Wei Liang, Yixin Zhu, and Siyuan Huang. Clone: Closed-loop whole-body humanoid teleoperation for long-horizon tasks. In Joseph Lim, Shuran Song, and Hae-Won Park, editors, *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pages 4493–4505. PMLR, 27–30 Sep 2025. URL https://proceedings.mlr.press/v305/li25h.html.

[31] Qiayuan Liao, Takara E. Truong, Xiaoyu Huang, Yuman Gao, Guy Tevet, Koushil Sreenath, and C. Karen Liu. Beyondmimic: From motion tracking to versatile humanoid control via guided diffusion, 2025. URL https://arxiv.org/abs/2508.08241.

[32] Chenhao Lu, Xuxin Cheng, Jialong Li, Shiqi Yang, Mazeyu Ji, Chengjing Yuan, Ge Yang, Sha Yi, and Xiaolong Wang. Mobile-television: Predictive motion priors for humanoid whole-body control. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5364–5371, 2025. doi: 10.1109/ICRA55743.2025.11128652.

[33] Zhengyi Luo, Jinkun Cao, Alexander Winkler, Kris Kitani, and Weipeng Xu. Perpetual humanoid control for real-time simulated avatars. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10861–10870, 2023. doi: 10.1109/ICCV51070.2023.01000.

[34] Zhengyi Luo, Ye Yuan, Tingwu Wang, Chenran Li, Sirui Chen, Fernando Castañeda, Zi-Ang Cao, Jiefeng Li, David Minor, Qingwei Ben, Xingye Da, Runyu Ding, Cyrus Hogg, Lina Song, Edy Lim, Eugene Jeong, Tairan He, Haoru Xue, Wenli Xiao, Zi Wang, Simon Yuen, Jan Kautz, Yan Chang, Umar Iqbal, Linxi ”Jim” Fan, and Yuke Zhu. Sonic: Supersizing motion tracking for natural humanoid whole-body control, 2025. URL https://arxiv.org/abs/2511.07820.

[35] H. Lv. Lafan1 retargeting dataset. https://huggingface.co/datasets/lvhaidong/LAFAN1_Retargeting_Dataset, 2025.

[36] Naureen Mahmood, Nima Ghorbani, Nikolaus F Troje, Gerard Pons-Moll, and Michael J Black. Amass: Archive of motion capture as surface shapes. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5442–5451, 2019.

[37] Jiageng Mao, Siheng Zhao, Siqi Song, Chuye Hong, Tianheng Shi, Junjie Ye, Mingtong Zhang, Haoran Geng, Jitendra Malik, Vitor Guizilini, and Yue Wang. Universal humanoid robot pose learning from internet human videos. In *2025 IEEE-RAS 24th International Conference on Humanoid Robots (Humanoids)*, pages 1–8, 2025. doi: 10.1109/Humanoids65713.2025.11203143.

[38] Gabriel B. Margolis and Pulkit Agrawal. Walk these ways: Tuning robot control for generalization with multiplicity of behavior. In Karen Liu, Dana Kulic, and Jeff Ichnowski, editors, *Proceedings of The 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pages 22–31. PMLR, 14–18 Dec 2023. URL https://proceedings.mlr.press/v205/margolis23a.html.

[39] Meta. Meta quest vr headsets and accessories — meta store, 2026. URL https://www.meta.com/quest/.

[40] Noboru Myers, Obin Kwon, Sankalp Yamsani, and Joohyung Kim. Child (controller for humanoid imitation and live demonstration): A whole-body humanoid teleoperation system. In *2025 IEEE-RAS 24th International Conference on Humanoid Robots (Humanoids)*, pages 1–6, 2025. doi: 10.1109/Humanoids65713.2025.11203119.

[41] Yixuan Pan, Ruoyi Qiao, Li Chen, Kashyap Chitta, Liang Pan, Haoguang Mai, Qingwen Bu, Hao Zhao, Cunyuan Zheng, Ping Luo, and Hongyang Li. Agility meets stability: Versatile humanoid control with heterogeneous data, 2025. URL https://arxiv.org/abs/2511.17373.

[42] Xue Bin Peng, Ze Ma, Pieter Abbeel, Sergey Levine, and Angjoo Kanazawa. Amp: adversarial motion priors for stylized physics-based character control. *ACM Trans. Graph.*, 40(4), July 2021. ISSN 0730-0301. doi: 10.1145/3450626.3459670. URL https://doi.org/10.1145/3450626.3459670.

[43] Amartya Purushottam, Jack Yan, Christopher Xu, and Joao Ramos. Heavy lifting tasks via haptic teleoperation of a wheeled humanoid. In *2025 IEEE-RAS 24th International Conference on Humanoid Robots (Humanoids)*, pages 345–350, 2025. doi: 10.1109/Humanoids65713.2025.11203084.

[44] Stephane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In Geoffrey Gordon, David Dunson, and Miroslav Dudík, editors, *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pages 627–635, Fort Lauderdale, FL, USA, 11–13 Apr 2011. PMLR. URL https://proceedings.mlr.press/v15/ross11a.html.

[45] Nikita Rudin, David Hoeller, Philipp Reist, and Marco Hutter. Learning to walk in minutes using massively parallel deep reinforcement learning. In *5th Annual Conference on Robot Learning*, 2021. URL https://openreview.net/forum?id=wK2fDDJ5VcF.

[46] Nikita Rudin, David Hoeller, Philipp Reist, and Marco Hutter. Learning to walk in minutes using massively parallel deep reinforcement learning. In Aleksandra Faust, David Hsu, and Gerhard Neumann, editors, *Proceedings of the 5th Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, pages 91–100. PMLR, 08–11 Nov 2022. URL https://proceedings.mlr.press/v164/rudin22a.html.

[47] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

[48] L. Sentis and O. Khatib. A whole-body control framework for humanoids operating in human environments. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pages 2641–2648, 2006. doi: 10.1109/ROBOT.2006.1642100.

[49] LUIS SENTIS and OUSSAMA KHATIB. Synthesis of whole-body behaviors through hierarchical control of behavioral primitives. *International Journal of Humanoid Robotics*, 02(04):505–518, 2005. doi: 10.1142/S0219843605000594. URL https://doi.org/10.1142/S0219843605000594.

[50] Yifan Sun, Rui Chen, Kai S. Yun, Yikuan Fang, Sebin Jung, Feihan Li, Bowei Li, Weiye Zhao, and Changliu Liu. SPARK: Safe protective and assistive robot kit. In *IFAC Symposium on Robotics*, 2025. URL https://intelligent-control-lab.github.io/spark/.

[51] Jie Tan, Tingnan Zhang, Erwin Coumans, Atil Iscen, Yunfei Bai, Danijar Hafner, Steven Bohez, and Vincent Vanhoucke. Sim-to-real: Learning agile locomotion for quadruped robots. In *Proceedings of Robotics: Science and Systems*, Pittsburgh, Pennsylvania, June 2018. doi: 10.15607/RSS.2018.XIV.010.

[52] Lujie Yang, Xiaoyu Huang, Zhen Wu, Angjoo Kanazawa, Pieter Abbeel, Carmelo Sferrazza, C. Karen Liu, Rocky Duan, and Guanya Shi. Omniretarget: Interaction-preserving data generation for humanoid whole-body loco-manipulation and scene interaction, 2025. URL https://arxiv.org/abs/2509.26633.

[53] Shiqi Yang, Minghuan Liu, Yuzhe Qin, Runyu Ding, Jialong Li, Xuxin Cheng, Ruihan Yang, Sha Yi, and Xiaolong Wang. ACE: A cross-platform and visual-exoskeletons system for low-cost dexterous teleoperation. In *8th Annual Conference on Robot Learning*, 2024. URL https://openreview.net/forum?id=7ddT4eklmQ.

[54] Yuxiang Yang, Xiangyun Meng, Wenhao Yu, Tingnan Zhang, Jie Tan, and Byron Boots. Continuous versatile jumping using learned action residuals. In Nikolai Matni, Manfred Morari, and George J. Pappas, editors, *Proceedings of The 5th Annual Learning for Dynamics and Control Conference*, volume 211 of *Proceedings of Machine Learning Research*, pages 770–782. PMLR, 15–16 Jun 2023. URL https://proceedings.mlr.press/v211/yang23b.html.

[55] Donghoon Youm, Hyunyoung Jung, Hyeongjun Kim, Jemin Hwangbo, Hae-Won Park, and Sehoon Ha. Imitating and finetuning model predictive control for robust and symmetric quadrupedal locomotion. *IEEE Robotics and Automation Letters*, 2023.

[56] Yanjie Ze, Zixuan Chen, Joao Pedro Araujo, Zi ang Cao, Xue Bin Peng, Jiajun Wu, and Karen Liu. TWIST: Teleoperated whole-body imitation system. In *9th Annual Conference on Robot Learning*, 2025. URL https://openreview.net/forum?id=htgNQHa6Ta.

[57] Yanjie Ze, Siheng Zhao, Weizhuo Wang, Angjoo Kanazawa, Rocky Duan, Pieter Abbeel, Guanya Shi, Jiajun Wu, and C. Karen Liu. Twist2: Scalable, portable, and holistic humanoid data collection system, 2025. URL https://arxiv.org/abs/2511.02832.

[58] Chong Zhang, Wenli Xiao, Tairan He, and Guanya Shi. Wococo: Learning whole-body humanoid control with sequential contacts, 2024.

[59] Tong Zhang, Boyuan Zheng, Ruiqian Nai, Yingdong Hu, Yen-Jen Wang, Geng Chen, Fanqi Lin, Jiongye Li, Chuye Hong, Koushil Sreenath, and Yang Gao. Hub: Learning extreme humanoid balance. In *9th Annual Conference on Robot Learning*, 2025. URL https://openreview.net/forum?id=FCpYuGtN4j.

[60] Ziwen Zhuang, Zipeng Fu, Jianren Wang, Christopher Atkeson, Sören Schwertfeger, Chelsea Finn, and Hang Zhao. Robot parkour learning. In *Conference on Robot Learning (CoRL)*, 2023.

[61] Ziwen Zhuang, Shenzhe Yao, and Hang Zhao. Humanoid parkour learning. In *8th Annual Conference on Robot Learning*, 2024. URL https://openreview.net/forum?id=fs7ia3FqUM.

Appendix

A. *Joint-Space Retarget Ablation*

Due to the page limit of the main paper, we provide a detailed discussion of the impact of joint-space retargeting on teleoperation performance in this section.

*1) **Latency**:* We benchmark the computational cost of different retargeting strategies. In our setting, joint-space retargeting targets the poses of the six robot links defined in Sec. II-A, whereas the vanilla GMR formulation [2] optimizes over 14 targets and is inevitably more expensive. All measurements are obtained on an Apple M4 processor (4.4 GHz), which is approximately 2–3× higher in clock frequency than the onboard CPUs commonly installed on Unitree robots (2.0 GHz for Jetson Orin NX and 1.7 GHz for Jetson Orin Nano). Consequently, the reported runtimes are expected to be 3–4× faster than those on the onboard hardware. All experiments are conducted on the $\mathcal{S}'_{\text{teleop}}$ dataset.

Strategy	**AVG**	**50%**	**90%**	**95%**	**99%**	**100%**
Cartesian-Space	**0.29**	**0.28**	**0.31**	**0.32**	**0.35**	**1.24**
Joint-Space (raw)	2.69	2.20	2.31	5.86	14.4	21.0
Joint-Space (fine)	7.34	6.10	11.5	12.5	18.2	31.5
Joint-Space (parallel)	13.1	13.2	19.6	20.2	25.5	40.5

TABLE IV: Ablation on retargeting time cost.

Joint-Space (raw) performs inverse kinematics (IK) sequentially on the six tracked links as defined in Sec. II-A, which differs slightly from conventional full-body human-to-humanoid retargeting settings that also track intermediate joints such as the elbows and knees. As a result, the inherent kinematic ambiguity can lead to implausible solutions. However, this reduced constraint set also lowers computational complexity: the average runtime of 2.9 ms indicates that the IK solver typically converges within only a few iterations, highlighting the efficiency and consistency of the proposed **Cartesian-Space** mapping.

To mitigate the irrational solutions produced by IK, we manually initialize the shoulder yaw, elbow, and wrist pitch joints at each IK iteration. This strategy substantially improves the stability of **Joint-Space (fine)** at the cost of increased computation time.

As discussed in Sec. II-A, the Cartesian-space mapping is inherently parallelizable. To ablate this property, instead of implementing a fully parallel retargeting pipeline, we reset all joint configurations (excluding the floating base) to zero at each IK iteration. This configuration renders **Joint-Space (parallel)** nearly parallelizable and provides a conservative estimate of the performance of a truly parallel retargeting approach.

*2) **Accuracy**:* In the main paper, we conclude that providing retargeted joint configurations improves policy accuracy when the retargeted robot link poses in $\mathrm{SE}(3)$ are already included in the observation and the joint configurations are added as auxiliary inputs. As shown in Table V, adding informative observations consistently improves performance regardless of the retargeting quality.

Tracking Error ↓	$E_{\text{l\_pos}}$	$E_{\text{l\_rot}}$	$E_{\text{l\_pos}}$	$E_{\text{l\_rot}}$	$E_{\text{l\_pos}}$	$E_{\text{l\_rot}}$
Observation	$[\mathbf{T}^r]$		$[\mathbf{T}^r, q_t]$		$[q_t]$	
Raw	0.062	0.22	0.059	0.22	0.113	0.53
Fine	0.062	0.22	0.055	0.20	**0.105**	**0.48**
Parallel	0.062	0.22	**0.052**	**0.19**	0.111	0.49

TABLE V: Ablations on observation. $E_{\text{l\_pos}}$ (m) represents the links tracking position error in pelvis frame; $E_{\text{l\_rots}}$ (rad) represents the links tracking rotation error in pelvis frame.

In contrast, when using only retargeted joint configurations, the end-effector tracking accuracy is substantially worse. **Providing link $\mathrm{SE}(3)$ poses significantly improves performance, while the retargeted joint configuration remains an optional enhancement that enables a trade-off between accuracy and latency.** Jointly considering the results in Tab. IV and Tab. V, we observe an approximate trade-off of 1 mm in tracking accuracy per 1 ms of additional latency (on an Apple M4 processor).

We note that all reported results are measured in simulation with joint configurations retargeted offline prior to execution; therefore, **any tracking errors induced by retargeting latency are not reflected in these metrics**.

B. *Optical Flow Latency Estimation*

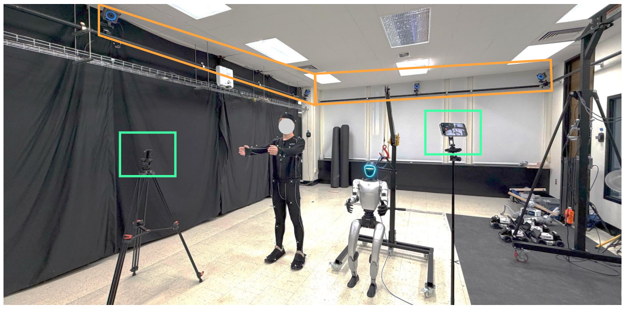


Fig. 7: Experimental setup for latency validation, with two cameras and an optical motion capture system simultaneously recording.

To demonstrate the accuracy and reliability of the proposed video-based latency estimation method, we conduct a controlled validation experiment using multi-view video recordings together with motion-capture-based ground-truth measurements. Specifically, we record a single reciprocating hand motion simultaneously using two cameras placed at different viewpoints, observing the same motion from distinct perspectives, as shown in Fig. 7. For each camera view, we independently apply the optical-flow-based pipeline described in Sec. VI-D to extract a one-dimensional motion signal and estimate the temporal offset between the human and robot motions, yielding two latency estimates.

In parallel, we attach optical motion capture markers to the robot hand, enabling direct access to the ground-truth Cartesian trajectories of the two end effectors, as shown in Fig. 8-(MoCap). Using these trajectories, we compute a reference latency by measuring the temporal offset between the corresponding motion signals derived from the motion capture data. This motion-capture-based estimate does not rely on image measurements or optical flow and therefore provides an independent ground truth for validation.

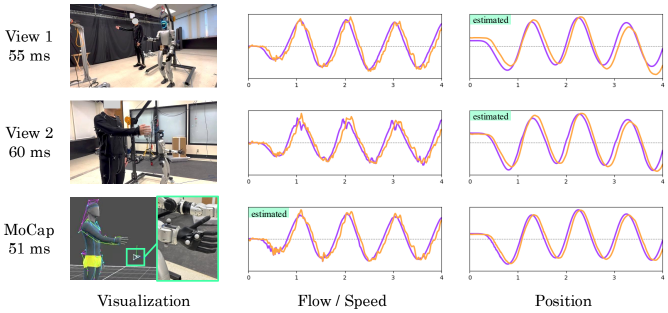


Fig. 8: Measured latencies of the same motion across different views and methods. Normalized optical-flow projections (or estimated velocities) and the corresponding position trajectories of the human and robot are shown on the right.

Fig. 8 presents the experimental results. For the two camera views, the middle column visualizes the projected optical flows, while the right column shows the accumulated displacement obtained by integrating the projected flow over time, which approximates the underlying motion trajectory. **The estimated latencies are 55 ms and 60 ms**, and their agreement serves as a consistency check for view invariance and robustness of the method. For the motion-captured end-effector positions, the middle column compares the estimated velocities, and the right column shows the directly tracked position waveforms. **The ground-truth latency calculated directly from the tracked positions is 51 ms**, which, together with the velocities computed from these positions, is consistent with the optical-flow-based estimates, confirming that the proposed method accurately captures end-to-end system latency. Minor discrepancies may arise due to non-ideal camera viewpoints or imperfect alignment between the chosen projection direction and the true motion. As illustrated in Fig. 8, the projected optical flow in View 1 closely aligns with the motion-capture-derived velocity, whereas the flow in View 2 is noticeably noisier, leading to a less accurate latency estimate. In addition, sub-frame-level variance (less than 16 ms for 60 FPS videos) can naturally occur due to measurement noise and temporal discretization effects.

C. Policy Learning Details

To clarify the policy learning formulation, we first introduce the notation used in this section. As defined in Sec. II-A, we denote the target pose of each tracking link as $\mathbf{T}^r_{\text{link}} = [\mathbf{R}^r_{\text{link}}, \mathbf{p}^r_{\text{link}}] \in \mathrm{SE}(3)$, and use $\mathbf{T}^r$ to represent the collection of all tracked link poses. The pose discrepancy between the measured and target link poses is denoted as $\Delta\mathbf{T}^r_{\text{link}} = [\Delta\mathbf{R}^r_{\text{link}}, \Delta\mathbf{p}^r_{\text{link}}]$. Notably, the pelvis link serves as the root link in simulation and as the IMU mounting base in the real robot; therefore, it is used to represent the global position and orientation of the robot.

Based on the foot link position $\mathbf{p}^r_{k,\text{foot}}$, we estimate the probability of foot–ground contact as follows:

$$\mathbb{P}_{\text{k\_foot}} = 1 - \min(1, \frac{\mathbf{p}^r_{\text{k\_foot,z}} - 0.2}{0.2} + \frac{||\dot{\mathbf{p}}^r_{\text{k}_\text{foot,xy}}|| - 0.2}{0.2})$$

Based on the estimated contact probability and the principle of momentum conservation, we approximate the contact force. We denote $\mathbb{F}$ as the proportion of the total body weight supported by each contact.

$$\mathbb{F}^t_{\text{k\_foot}} = \mathbb{P}^t_{\text{k\_foot}} / \mathbb{E}[\mathbb{P}^{t-15:t+15}_{\text{left\_foot}} + \mathbb{P}^{t-15:t+15}_{\text{right\_foot}}]$$

where $\mathbb{E}$ denotes a weighted sum with a quadratic weighting function that vanishes at both endpoints. We use the **Joint-Space (fine)** setting in Table IV to obtain the reference joint configuration q_t at each frame, where the subscript t denotes the target configuration.

1) Observation: The future interpretation function $\mathscr{I}$ is defined as follows:

Future pelvis translation	$\mathbf{p}^r_{\text{pelvis,t:t+H}} - \mathbf{p}^r_{\text{pelvis,t}-1}$
Future pelvis rotation	$\mathbf{r}^{r,\top}_{\text{pelvis,t:t+H}} \cdot \mathbf{r}^r_{\text{pelvis,t}-1}$
Link poses	$\mathbf{T}^{r'}_{t:t+H}$
Link velocities (privilege)	$\mathbf{V}^r_{t:t+H}$
Retargeted joint configuration (privilege)	$q_t, \dot{q}_t$
Foot contact probability	$\mathbb{P}_{\text{k\_foot}}$

The proprioception observation $\mathbf{o}^{\text{proprio}}$ is listed below:

Last action	a_{t-1}
Joint configuration	$q, \dot{q}$
Pelvis rotation	$\mathbf{r}_{\text{pelvis}}$
Pelvis linear velocity (privilege)	$\dot{\mathbf{p}}_{\text{pelvis}}$
Pelvis angular velocity	$\dot{\mathbf{r}}_{\text{pelvis}}$

Additional privileged observation $\mathbf{o}^{\text{priv}}$ is listed below:

Domain Randomization Parameters	–
Residual joint configuration	$\Delta q, \Delta\dot{q}$
Residual link poses	$\Delta\mathbf{T}^r$
Foot contact force	$\mathbb{F}^{\text{target}}_{\text{k\_foot}}, \mathbb{F}^{\text{measure}}_{\text{k\_foot}}$

2) Reward: The exact reward formulation involves axis- and link-specific weighting terms and is therefore omitted for brevity. Below, we present the simplified primary reward components.

Reward Term	**Expression**	**Scale**
Global tracking link poses	$-\lvert\lvert(\mathbf{U}^r)^{-1}\mathbf{T}^r\rvert\rvert^2_2$	30
Local tracking link poses	$-\lvert\lvert(\mathbf{U}^{r'})^{-1}\mathbf{T}^{r'}\rvert\rvert^2_2$	20
Retargeted DoF position	$-\lvert\lvert q_t - q\rvert\rvert^2_2$	3
Retargeted DoF velocity	$-\lvert\lvert \dot{q}_t - \dot{q}\rvert\rvert^2_2$	0.02
Foot contact reward	$-\left(\mathbb{F}^{\text{target}}_{\text{k\_foot}} - \mathbb{F}^{\text{measure}}_{\text{k\_foot}}\right)^2$	3
Foot contact penalty	$-\left(\mathbb{I}[\mathbb{P}_{\text{k\_foot}} < 0.2] \cdot \mathbb{F}^{\text{measure}}_{\text{k\_foot}}\right)^2$	10
Torque Penalty	$-\lvert\lvert\tau\rvert\rvert^2_2$	0.0001
Action rate	$-\lvert\lvert a_{t-1} - a_t\rvert\rvert^2_2$	0.2

3) Domain Randomization: Domain randomization is applied to motor dynamics, contact friction, and torso mass distribution. For motor dynamics, we apply four randomizations:

$$\tau = \alpha_{\text{strength}}(\alpha_{k_p} k_p(q_t - q + \beta_{\text{offset}}) + \alpha_{k_d}(\eta\,\dot{q}_t - q_t))$$

Detailed distributions are as follows:

k_p ratio α_{k_p}	$\mathcal{U}(0.8, 1.2)$
k_d ratio α_{k_d}	$\mathcal{U}(0.8, 1.2)$
Motor strength α_{strength}	$\mathcal{U}(0.8, 1.2)$
Motor offset β_{offset}	$\mathcal{U}(-0.1, 0.1)$
Friction ratio	$\mathcal{U}(0.3, 1.0)$
Torso added mass (kg)	$\mathcal{U}(-2, 5)$
Torso CoM displacement (m)	$\mathcal{U}(-0.05, 0.05)^3$

4) Policy Learning: All actor and critic networks are implemented as MLPs with hidden dimensions [1024, 512, 256] and ReLU as activation layer. We adopt an adaptive learning rate schedule in PPO with a target KL divergence of 0.01. The remaining PPO hyperparameters are listed below:

γ	0.99
λ_{GAE}	0.95
Entropy coefficient	0.003
Value loss coefficient	1.0
Rollout length	24
Optimizer	ADAM
# epoch	5
# mini batch	8
# iteration	6000

We utilize DAgger with the following hyperparameters:

learning rate η	0.0003
Batch size	256
Rollout length	24
Optimizer	ADAM
# epoch	10
# iteration	1500