2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

H-Zero: Cross-Humanoid Locomotion Pretraining Enables Few-shot Novel Embodiment Transfer

Yunfeng Lin[1,2], Minghuan Liu[3,†], Yufei Xue[1], Ming Zhou[2], Yong Yu[1], Jiangmiao Pang[2], Weinan Zhang[1,2,*]
[1]Shanghai Jiao Tong University [2]Shanghai AI Lab [3]ByteDance Seed

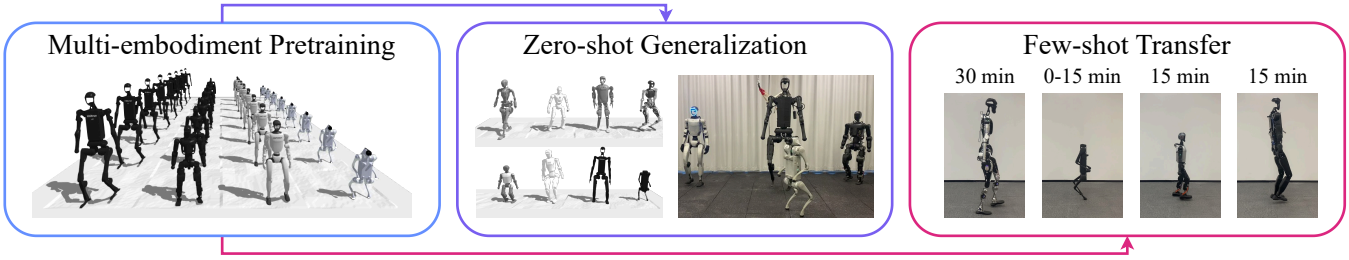

Fig. 1: **Left:** We propose a locomotion pretraining pipeline for humanoids by mixing multiple randomized embodiments into the training set. **Middle:** The pretrained policy shows moderate adaptability to unseen embodiments and real hardware. **Right:** Fine-tuning the pretrained policy achieves stable control on unseen robots with minimal additional training time.

***Abstract*— The rapid advancement of humanoid robotics has intensified the need for robust and adaptable controllers to enable stable and efficient locomotion across diverse platforms. However, developing such controllers remains a significant challenge because existing solutions are tailored to specific robot designs, requiring extensive tuning of reward functions, physical parameters, and training hyperparameters for each embodiment. To address this challenge, we introduce H-Zero, a cross-humanoid locomotion pretraining pipeline that learns a generalizable humanoid base policy. We show that pretraining on a limited set of embodiments enables zero-shot and few-shot transfer to novel humanoid robots with minimal fine-tuning. Evaluations show that the pretrained policy maintains up to 81% of the full episode duration on unseen robots in simulation while enabling few-shot transfer to unseen humanoids and upright quadrupeds within 30 minutes of fine-tuning.**

I. Introduction

The rapid proliferation of novel and customized humanoid robot designs has intensified the demand for control algorithms capable of enabling robust and versatile locomotion across diverse embodiments. Humanoid robots, characterized by their anthropomorphic structure and high degrees of freedom (DoFs), offer unparalleled versatility in tasks such as bipedal walking, dynamic balancing, and complex manipulation. However, this anthropomorphic design also introduces significant challenges in controller development because of the intricate interplay of high-dimensional joint configurations, variable morphologies, and dynamic physical interactions.

Deep reinforcement learning (DRL) methods have shown promise in training locomotion controllers by leveraging physical simulations and trajectory sampling. Combined with techniques such as domain randomization [1], knowledge distillation [2], [3], explicit models [4], [5], and curriculum learning [6], policies trained in simulation can consistently transfer to the real world while maintaining performance, efficiency [7] and safety [8]. However, most existing approaches treat each robot variant as a distinct learning problem, resulting in policies tightly coupled to specific morphologies. Even minor variations in joint layouts, mass distributions, or limb dimensions often degrade performance, requiring resource-intensive retraining from scratch. As the diversity of humanoid platforms continues to grow, there is an urgent need for generalizable controllers that can adapt to new embodiments with minimal effort.

To address this challenge, we propose H-Zero, a novel pretraining pipeline for developing generalized locomotion policies for humanoid robots. Our approach introduces transformation layers at the policy's input and output to standardize control semantics across diverse humanoid embodiments, enabling unified policy representations. We further incorporate cross-embodiment diversity through randomized physical parameters, varied policy observations, diverse environment rollouts, and exploratory learning strategies. By integrating these techniques, H-Zero learns a robust base policy capable of controlling multiple humanoid models simultaneously. This pretrained policy can serve as a foundation for few-shot adaptation, allowing rapid fine-tuning to novel hardware with minimal data and computational resources.

We evaluate H-Zero on a comprehensive suite of humanoid robots with varying kinematic structures and physical properties, including simulated and real-world platforms. Our experiments demonstrate that the pretrained policy achieves

†Minghuan did the work when at ByteDance Seed.
*Corresponding author.

superior few-shot adaptation, matching the performance of policies trained from scratch in only hundreds of epochs across benchmark locomotion tasks. Moreover, both the pretrained and fine-tuned policies exhibit consistent sim-to-real transfer, enabling robust performance on physical robots and highlighting the feasibility of shared policy learning for scalable humanoid control.

In summary, the contributions of this paper are:

- A unified control interface that employs state transformations to standardize policy inputs and outputs across diverse humanoid embodiments.
- A cross-embodiment training environment with physical randomization and various training strategies to learn generalizable locomotion policies from a set of existing robots.
- Demonstrating that cross-embodiment locomotion pretraining enables zero-shot and few-shot adaptation to novel robots, significantly reducing the need for extensive retraining.

II. Related Work

A. Legged locomotion via reinforcement learning

Reinforcement learning provides a competent alternative for developing robot locomotion controllers by optimizing policies against task objectives instead of relying on manual implementation. RL has been widely applied to various robots and tasks, including quadrupeds [9], [10], [2], [11], wheeled robots [12], and humanoids [13], [14], [15], [16]. For quadrupeds, agile locomotion has been achieved over various terrains with both blind [17], [18], [19], [20] and exteroception-based RL policies [3], [21]. For humanoid robots, RL also enables whole-body control that achieves diverse gaits [22], motion mimics [23], [24], [25], as well as integrations with upper-body teleoperation and manipulation [26], [27].

B. Cross-embodiment locomotion learning

With the development of new robot models, recent research focuses on synthesizing controllers that generalize to hardware with varied shapes, weights, and even morphologies. Some works propose to adopt specialized network architectures as a form of inductive bias. For example, Graph Neural Networks (GNNs) can capture the robot's morphology by modeling a fixed set of joints as graph nodes [28], [29]. On the other hand, transformer-based methods allow flexible observation and action dimensions by treating them as sequences [30], [31], [32]. This enables unified control across varied dimensions but demands high computational resources.

The combination of RL with other generative models such as diffusion [33] has also been explored. Beyond legged robots, cross-embodiment methods have been applied to dexterous robot hands [34] to enable embodiment-aware manipulation.

Another line of work such as GenLoco procedurally generates embodiments with randomized morphological and physical properties [35], [36], covering quadrupeds, hexapods, and humanoids. This improves policy generalization as a form of domain randomization [37], [38], [39]. However, determining the distribution of generated embodiments is crucial for policy performance and requires considerable expert knowledge for refinement. As a result, real-robot applications remain challenging, especially for humanoids because of their control complexity. In this work, we propose a locomotion pretraining framework for humanoids that enables rapid adaptation to real robots using less training time and fewer hyperparameters. Unlike GenLoco, which depends on a predefined quadruped template, our approach interoplates between real-world robot models with targeted domain randomization, reducing hyperparameters complexity and improving transfer reliability.

III. Preliminaries

Humanoid locomotion control seeks to develop policies for stable and versatile movement in anthropomorphic robots, including walking, running, turning, and navigating complex terrain. Reinforcement learning (RL) is commonly used to optimize such policies by maximizing expected cumulative reward, learning a policy $\pi(o, c)$ that maps observations and commands to joint-level actions. The reward function needs to balance multiple objectives: tracking velocity commands (e.g., forward, lateral, and yaw), maintaining base height, and aligning torso orientation. Additional terms penalize energy inefficiency, instability, or gait deviations to promote robust and efficient motion.

However, most RL-based controllers are morphology-specific, requiring tailored reward functions for each robot. Our framework addresses this by learning a generalizable base policy across embodiments, enabling few-shot adaptation to novel morphologies.

IV. Cross-Embodiment Humanoid Control

To support generalization across diverse humanoid embodiments, we augment the RL state space $\mathcal{S}$ to include embodiment-specific parameters:

$$\mathcal{S} = \mathcal{Q} \times \mathcal{E} \ , \tag{1}$$

where $\mathcal{Q}$ captures the system's kinematic states (e.g., base position, joint angles, velocities), and $\mathcal{E}$ represents embodiment-specific parameters such as joint layout, link dimensions, and inertial properties. During training, an embodiment $\boldsymbol{e} \in \mathcal{E}$ is sampled per rollout and held fixed, exposing the policy to a range of morphologies.

The observation space $\mathcal{O}$ similarly includes both kinematic state and embodiment descriptors. The objective is to learn a unified policy that generalizes across embodiments by capturing shared control strategies.

A. Unified Control Semantics

Learning policies for individual humanoid robots often relies on embodiment-specific joint configurations and control interfaces, resulting in incompatible action and observation spaces and poor transferability. To address this, we introduce a unified control representation that standardizes joint semantics across robots, enabling consistent policy learning and deployment.

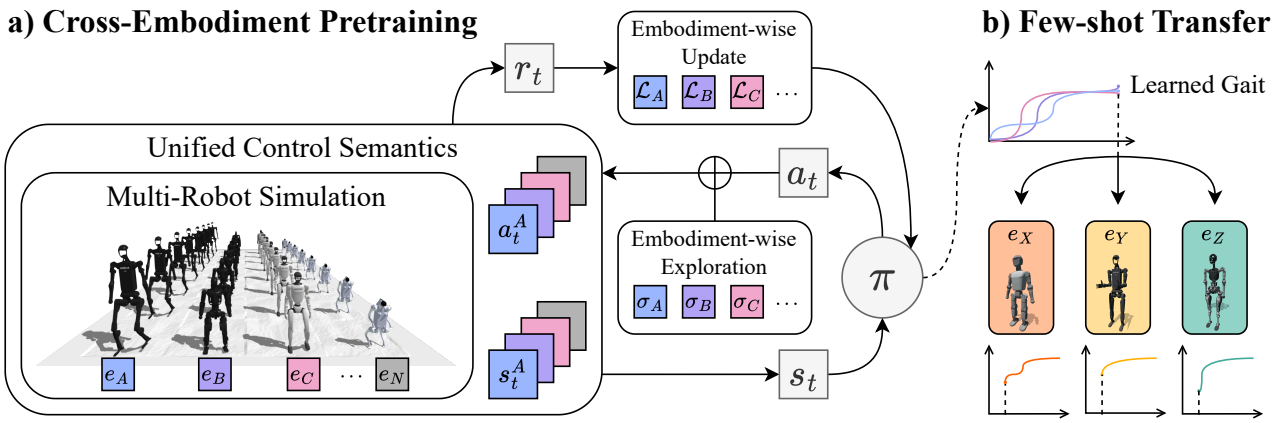

Fig. 2: **Method overview. a)** The policy is pretrained by learning on a diverse set of humanoid embodiments through multi-robot simulation with unified control. Training progress is dynamically balanced with embodiment-wise exploration and gradient updates. **b)** At deployment, the pretrained policy supports few-shot adaptation to novel robots.

We define a hardware-agnostic joint state space comprising current positions, velocities, and target positions for key rotational joints (e.g., head, shoulders, knees, ankles). Each robot maps its physical joints to this space based on kinematic roles. For example, we designate the **wrist pitch** joint as the one rotating the hand link around the Y-axis from a reference pose.

This mapping is formalized as a bidirectional transformation between the robot's physical joint space and the unified environment space:

$$\boldsymbol{q}_{\text{env}} = \boldsymbol{M} \cdot \boldsymbol{q}_{\text{phy}}, \quad \boldsymbol{q}_{\text{phy}} = \boldsymbol{M}^T \cdot \boldsymbol{q}_{\text{env}} \ , \tag{2}$$

where $\boldsymbol{M}$ encodes index assignments and zero-padding to accommodate varying degrees of freedom (DoFs) [40], [41], [33].

To ensure consistent motion across embodiments, we further apply kinematic alignment:

$$\boldsymbol{q}_{\text{env}} = \boldsymbol{M} \cdot (\boldsymbol{s} \odot (\boldsymbol{q}_{\text{phy}} - \boldsymbol{b})) \ , \tag{3}$$

where $\boldsymbol{s}$ adjusts joint directions so that rotational axes conform to a right-hand frame, and $\boldsymbol{b}$ shifts joint neutral positions into a standardized upright pose.

Additional properties are normalized as well: rigid-body contacts are reordered to ensure correct penalty terms, IMU readings are transformed into a reference frame, and the robot base is always mapped to the pelvis link. These transformations ensure consistent physical motions and sensor interpretations across embodiments.

B. Embodiment Descriptors

The unified control semantics introduced above enable cross-embodiment control by standardizing the input and output spaces of a single policy with fixed parameter dimensions. Typical simulation environments, however, are designed to train standalone policies tailored to individual robot variants, making them inherently hardware-specific. In such settings, controllers are informed solely with proprioceptive observations such as joint positions, velocities, and inertial measurement unit (IMU) readings, without explicit awareness of the robot's configuration because these physical properties are assumed as static and implicit.

In contrast, a cross-embodiment policy must dynamically adapt to the physical properties of the controlled robot to generate optimal actions for stable and efficient locomotion. To achieve embodiment-aware control, we introduce embodiment descriptors: compact, informative features that encode key aspects of each robot's physical properties. These descriptors span multiple domains, including kinematics (*e.g.*, joint limits, rotation axes), topology (*e.g.*, hierarchical joint structures), geometry (*e.g.*, link lengths, shapes), and dynamics (*e.g.*, mass, inertia, stiffness, damping). The resulting joint and rigid body features are arranged according to the unified space in Sec. IV-A to ensure consistent semantics across robots.

$$\begin{aligned} Z_{\text{JD}}(\boldsymbol{e}) &= \boldsymbol{M} \cdot [(K_p, K_d, \tau_{max})_{1\ldots n}]^T \\ Z_{\text{RD}}(\boldsymbol{e}) &= \boldsymbol{M} \cdot [(m, h, I)_{1\ldots n}]^T \\ Z_{\text{e}}(\boldsymbol{e}) &= [Z_{\text{JD}}(\boldsymbol{e}), Z_{\text{RD}}(\boldsymbol{e}), Z_{\text{Kine}}(\boldsymbol{e}), Z_{\text{Geom}}(\boldsymbol{e})] \end{aligned} \tag{4}$$

In the evaluation section, we compare providing embodiment descriptors to the policy as observable inputs versus only to the critic as privileged information in an asymmetric actor-critic architecture [42].

C. Embodiment Training Set

To train a cross-embodiment policy capable of controlling multiple humanoid robots, we generate simulated trajectories using a diverse set of robot models, leveraging unified control semantics and embodiment descriptors. The policy is optimized to perform locomotion tasks across all embodiments concurrently.

While prior work applies domain randomization (DR) to perturb system parameters such as joint friction and mass [43], [44], [45], we extend this technique by broadening the DR range to better approximate the dynamics of varied robot morphologies. Specifically, we double the perturbation bounds

for rigid-body and joint dynamics, while structural diversity (e.g., geometry and topology) is introduced by mixing existing robot models.

Unlike approaches that procedurally generate randomized hardware from scratch [36], which often require extensive tuning of physical hyperparameters, our method relies on targeted extrapolation from real-world designs. This yields strong generalization performance with minimal engineering overhead, as demonstrated in our experiments.

D. Embodiment-Aware Learning

We adopt a unified reward formulation across all robots, following the design in [14]. While the structure of the reward function remains identical, its coefficients such as nominal base height and stance width are tailored to each embodiment. This multi-robot setup presents greater learning challenges than single-robot training due to diverse dynamics, stability profiles, and task complexities.

To address this, we introduce an embodiment-aware training strategy that assigns independent exploration sampling variances σ to each robot type. These variances are updated adaptively based on gradients from each embodiment, enabling tailored exploration: robots exhibiting slower learning progress receive higher variance to promote broader action sampling, while more stable embodiments converge under reduced variance.

In addition to adaptive exploration, we dynamically reweight the loss of each embodiment type based on their current performance, quantified by average episodic return.

$$\mathcal{L}_{\text{total}} = \sum_{i=1}^{N} w_i \mathcal{L}_i = \sum_{i=1}^{N} (1 - \frac{R_i - R_{\min}}{R_{\max} - R_{\min} + \epsilon}) \mathcal{L}_i \quad (5)$$

This dynamic loss scheduling ensures that underperforming embodiments receive increased gradient emphasis during training. Together, these techniques promote balanced learning progress across the embodiment set, accelerating overall convergence and enhancing policy robustness across a wide range of humanoids and even standing quadrupeds.

E. Cross-Embodiment Transfer

Leveraging a specialized learning environment and task-aware training algorithm, H-Zero learns a family of humanoid locomotion policies that generalize across robot variants, serving as a pretrained controller for rapid transfer.

The framework begins by curating a diverse set of robot embodiments with varying structures (e.g., joint layouts, limb proportions) and dynamics (e.g., mass, inertia). During pretraining, the policy learns to control these embodiments using extended domain randomization, unified semantics, and embodiment descriptors, yielding a robust base policy.

To transfer to a novel robot, we perform few-shot fine-tuning using a small set of trajectories. A key design is to use a low action variance during initial sampling, as excessive noise can destabilize walking behaviors and hinder adaptation. This approach supports zero-shot inference and few-shot adaptation on similar morphologies, and enables sim-to-real deployment.

TABLE I: Robot models used in our experiments

Model Name	Alias	DoFs			Mass	Base Height
		Arm	Waist	Leg		
Unitree [46] H1 Gen1	H1	4	1	5	51.6	0.98
Unitree G1 EDU	G1	7	3	6	33.3	0.78
PND [47] Adam Lite	Adam	5	3	6	58.4	0.88
Fourier [48] GR-1 Pro	GR1	7	3	6	56.9	0.91
Fourier N1	N1	5	1	6	39.7	0.68
Booster T1 [49]	T1	4	1	6	31.6	0.65
EngineAI PM01 [50]	PM01	5	1	6	40.9	0.82
OpenLoong [51]	OGHR	4	1	6	75.9	1.12
Leju Kuavo S42 [52]	Kv	7	0	6	56.9	0.85
Dobot Atom [53]	Atom	2	0	6	58.8	0.88
Unitree H1 Gen2	H1-2	4	1	5	67.3	0.96
Unitree Go2	Go2	3	0	3	15.0	0.54
Unitree A1	A1	3	0	3	12.4	0.48
Unitree Aliengo	AGo	3	0	3	24.9	0.60

In experiments, H-Zero outperforms baselines trained from scratch in both task performance and learning efficiency on unseen robots.

V. Evaluations

We focus on three research questions through simulation and real-world evaluations:

1) How does the performance of the pretrained policy scale with the number and diversity of training embodiments?
2) How efficiently does the pretrained policy adapt to novel embodiments in few-shot settings?
3) How well does the policy generalize to real-world deployment across different hardware platforms?

Network architecture. We employ a simple multilayer perceptron (MLP) with ELU [55] activation for the actor and critic network, which takes as input a five-step history of observations. A state estimator predicts the base linear velocity to improve command tracking [56].

Training setups and robots involved. Training is conducted using the Isaac Gym [57] physics simulator, running on a single NVIDIA GeForce RTX 4090 GPU. The x-y-yaw velocity commands are randomly sampled within the ranges: $v_x \in [-0.6, 1.2], v_y \in [-0.4, 0.4], v_\psi \in [-1.0, 1.0]$, with a curriculum that begins at half these ranges to facilitate early learning. The set of robot embodiments used for training and evaluation is summarized in Table I.

Metrics. To quantify locomotion performance across diverse humanoid embodiments, we use the following metrics to assess the policy's ability to maintain stable and accurate motion across different embodiments:

- Normalized episode length: the ratio of executed policy steps to the episode limit, indicating transferability and behavioral stability on novel robots.
- E_{v_x}, E_{v_y}: error in the robot's root linear velocity across the horizontal plane, measuring deviation from commanded forward and lateral velocities.
- E_{v_ψ}: error in the root angular velocity about the z-axis, evaluating the accuracy of turning motions.

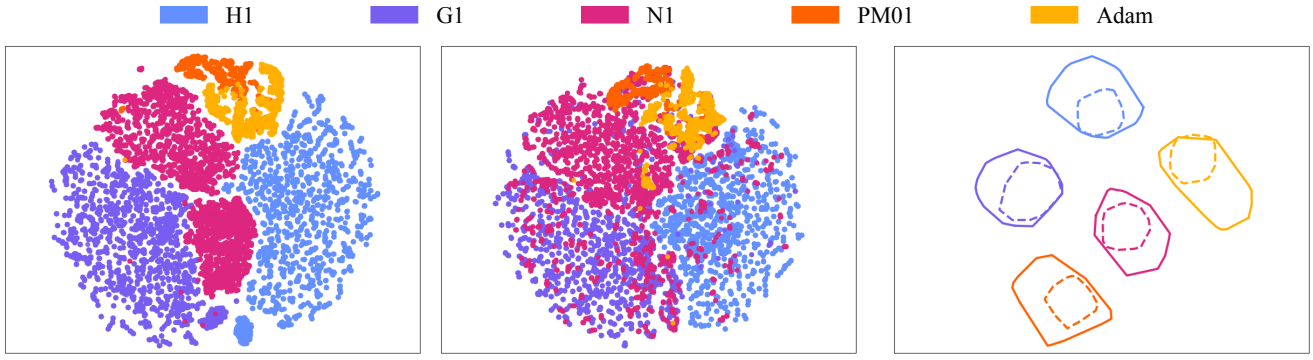

Fig. 3: **t-SNE [54] visualization of rollout trajectories and embodiment descriptors under different domain randomization (DR). Left**: standard DR for single-robot training. **Right**: extended DR proposed in Sec. IV-C in cross-embodiment pretraining. With extended DR, trajectories from unseen robots without randomization overlap with the broadened training distribution, demonstrating improved transferability.

Fig. 4: t-SNE convex hulls of embodiment parameters under standard (dashed) and quadrupled (solid) DR ranges, which retain clear boundaries even under strong randomization.

TABLE II: Cross-embodiment generalization performance of pretrained policies

Settings	Evaluation Robot														Test
	H1	N1	G1	GR1	PM01	Kv	Adam	T1	OGHR	Atom	H1-2	Go2	A1	AGo	Mean
Ablation A: Training Set															
H1	1.00	0.07	0.08	0.26	0.26	0.21	0.07	0.07	0.28	0.04	0.99	0.10	0.12	0.03	0.20
H1,N1	1.00	1.00	1.00	**0.67**	0.87	0.90	0.12	**0.99**	0.41	1.00	0.99	0.12	0.12	0.03	0.60
H1,N1,GR1	1.00	1.00	1.00	0.84	0.82	0.83	0.78	0.54	**0.73**	1.00	0.99	0.11	0.13	0.03	0.63
H1,N1,G1	1.00	0.99	1.00	0.58	**0.99**	0.54	**0.85**	0.48	0.71	**1.00**	0.87	0.15	0.14	0.03	0.58
H1,N1,G1,A1	1.00	0.99	0.99	0.48	0.91	0.86	0.17	**0.99**	0.71	0.97	**1.00**	**0.99**	0.99	0.70	**0.81**
H1,N1,G1,GR1,Go2	1.00	1.00	1.00	1.00	0.58	**1.00**	0.36	0.72	0.64	0.85	0.95	0.95	**0.99**	**0.99**	0.79
Ablation B: Domain Randomization															
Single-robot range	1.00	1.00	1.00	0.29	0.26	0.52	0.76	0.21	0.59	0.99	0.55	0.13	0.13	0.03	0.53
Quadrupled range	1.00	1.00	1.00	0.47	0.16	0.82	0.11	**1.00**	0.61	0.85	0.92	0.11	0.10	0.07	0.47
Ablation C: Embodiment Descriptors															
No Descriptor	1.00	1.00	1.00	**0.68**	0.25	0.70	0.09	0.38	0.23	0.99	0.98	0.14	0.14	0.03	0.54
Observable Desc.	1.00	1.00	1.00	0.25	0.29	0.10	0.07	0.62	0.18	0.08	0.99	0.08	0.09	0.02	0.41
Ablation D: Action space size															
Whole-body (32)	1.00	1.00	1.00	0.24	0.51	0.44	0.73	0.11	0.47	0.70	0.76	0.09	0.09	0.08	0.56
Legs + Waist (15)	1.00	1.00	1.00	0.42	**0.99**	0.76	0.08	0.62	0.03	**1.00**	0.96	0.11	0.03	0.09	0.46

Tracking errors are reported as mean values per episode, excluding the first 50 steps to avoid transient initialization effects.

A. *Cross-embodiment pretraining*

To assess the role of embodiment diversity in pretraining, we compare policies trained on different subsets of robot models. Training is conducted across 8192 parallel environments, evenly distributed among the selected embodiments. All policies are trained for 50,000 epochs using the same reward functions, taking approximately 24 hours. Unless otherwise noted, the default training set consists of H1, N1, and G1; the action space is limited to 12 leg joints; the randomization range is doubled relative to single-robot settings; and embodiment descriptors are provided only to the critic. Evaluations are conducted at 60% of the velocity command ranges across 2048 environments for four episodes, with domain randomization disabled to isolate effects from embodiment differences. Results are summarized in Table II and analyzed below.

Embodiment training set. We vary the composition of the training set to study the effect of embodiment mixing, as detailed in Sec. IV-C. Results in Table II show that policies trained on a broader set of embodiments consistently achieve higher performance on unseen robots than those trained on fewer embodiments, especially when the training set includes morphologically distinct robots. We further validate the effectiveness of our training strategies in Sec. IV-D. As visualized in Figure 5, dynamically adjusting sampling variances and loss scales across embodiments leads to more balanced learning progress.

Domain randomization. Ablation B confirms that expanding

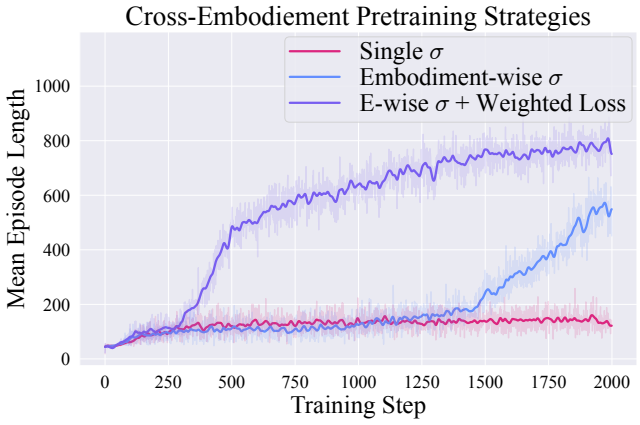

Fig. 5: Mean episode length of cross-embodiment trainings (H1, N1, G1, and A1) under different training strategies.

the domain randomization range improves transferability to unseen robots. To visualize this effect, we collect the trajectories of pretrained policy rollouts under different randomization settings, and visualize them in Figure 3. Each point represents a five-step state sequence sampled from a successful trajectory of one specific embodiment. The first three robot models from the training set are randomized, whereas the last two unseen robots use no randomization. Under single-embodiment randomization, trajectories from unseen robots show little similarity to the training set. Extending the randomization range expands the coverage and begins to overlap with unseen robots, indicating better transferability across embodiments.

Embodiment descriptor. Ablation C shows that policies using observable embodiment descriptors perform worse than those relying on privileged descriptors. To understand this, we visualize the extracted system parameters in Figure 4. Despite randomization, the parameters form disjoint clusters per robot, which is expected since the embodiment geometry and structure are not randomized. This also suggests that similar physical effects may arise from distinct configurations (e.g., higher torque compensating for increased mass).

Action space size. Expanding the action space to include waist and upper-body control has a negative impact on performance. This is likely due to increased dimensionality and weak reward coupling for upper-body joints, signaling the need for more targeted objectives and regularization.

B. Few-shot transfer

We assess the transferability of pretrained policies to unseen robot models. Specifically, we compare the adaptation performance of policies initialized from pretrained weights versus those trained from scratch. Thanks to the unified action and observation spaces, weight reuse is straightforward. Transfer performance is evaluated based on cumulative reward and tracking error after a fixed number of training epochs, as shown in Table III. Training configurations are denoted as a single letter followed by the number of training epochs onward: S indicates training from scratch, while P denotes fine-tuning from a pretrained policy. For example, P 0 refers to direct evaluation of the pretrained policy without additional training.

TABLE III: Per-embodiment tracking performance

Robot	Training Epochs		Return ↑	E_{v_x} ↓ (cm/s)	E_{v_y} ↓ (cm/s)	E_{v_ϕ} ↓ (deg/s)
Adam	S	20k	53 ± 1.4	4.6 ± 2.9	6.9 ± 4.4	6.0 ± 0.9
	S	50k	52 ± 1.2	3.9 ± 2.0	**5.6 ± 2.1**	**3.1 ± 0.5**
	P	0	20 ± 5.6	22 ± 19	13 ± 5.6	13 ± 11
	P	1k	49 ± 1.8	4.8 ± 3.3	8.1 ± 3.0	4.3 ± 0.8
	P	2k	**52 ± 1.5**	**3.7 ± 3.0**	7.2 ± 2.1	3.5 ± 0.5
T1	S	25k	48 ± 2.5	9.0 ± 8.9	10 ± 2.5	7.6 ± 1.0
	S	50k	49 ± 2.3	8.8 ± 8.4	12 ± 2.9	7.8 ± 0.6
	P	0	37 ± 5.6	20 ± 7.6	16 ± 3.6	27 ± 4.0
	P	500	50 ± 2.9	13 ± 10	10 ± 2.4	**5.8 ± 0.4**
	P	2k	**52 ± 1.9**	**8.2 ± 8.0**	10 ± 2.7	6.5 ± 0.4
H1-2	S	25k	47 ± 6.2	14 ± 17	8.9 ± 4.1	5.3 ± 1.8
	S	50k	51 ± 1.0	5.7 ± 1.7	**7.8 ± 1.0**	4.7 ± 0.8
	P	0	31 ± 3.2	17 ± 5.8	10 ± 1.1	8.0 ± 0.3
	P	500	45 ± 4.5	5.2 ± 3.4	9.7 ± 1.8	5.5 ± 0.8
	P	5k	**53 ± 1.1**	**4.3 ± 1.8**	8.4 ± 1.3	**4.3 ± 0.4**
AGo	S	25k	50 ± 2.7	18 ± 12	12 ± 6.3	18 ± 4.2
	S	50k	53 ± 2.8	17 ± 11	**12 ± 6.2**	14 ± 3.8
	P	0	53 ± 3.0	**11 ± 6.1**	13 ± 6.4	13 ± 1.6
	P	500	53 ± 3.0	13 ± 8.4	12 ± 6.7	**13 ± 1.8**

Notation: S = Scratch initialization; P = Pretrained initialization; Epochs indicate retraining/fine-tuning duration.

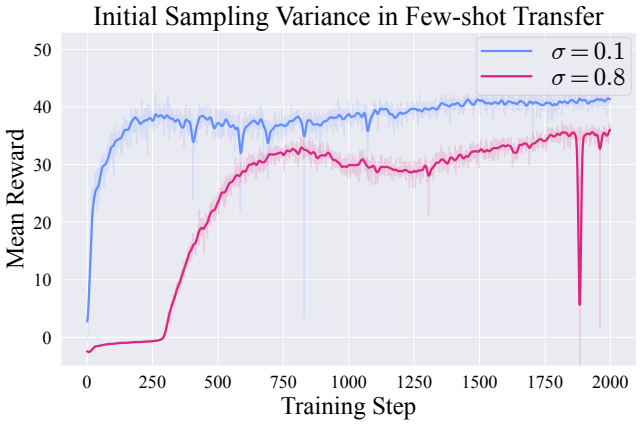

Fig. 6: Few-shot transfer reward curves of PND Adam robot under different initial action sampling variance.

The results show that pretrained policies can adapt to novel embodiments within hundreds of training epochs, outperforming policies trained from scratch with thousands of epochs. This efficiency holds across different robot morphologies, highlighting the scalability of our approach.

In Figure 6, we further validate the impact of initial sampling variance during transfer learning. Specifically, we show that using a reduced action variance in early trajectory collection leads to much faster adaptation compared to default training settings by preserving valid locomotion behaviors from the pretrained policy.

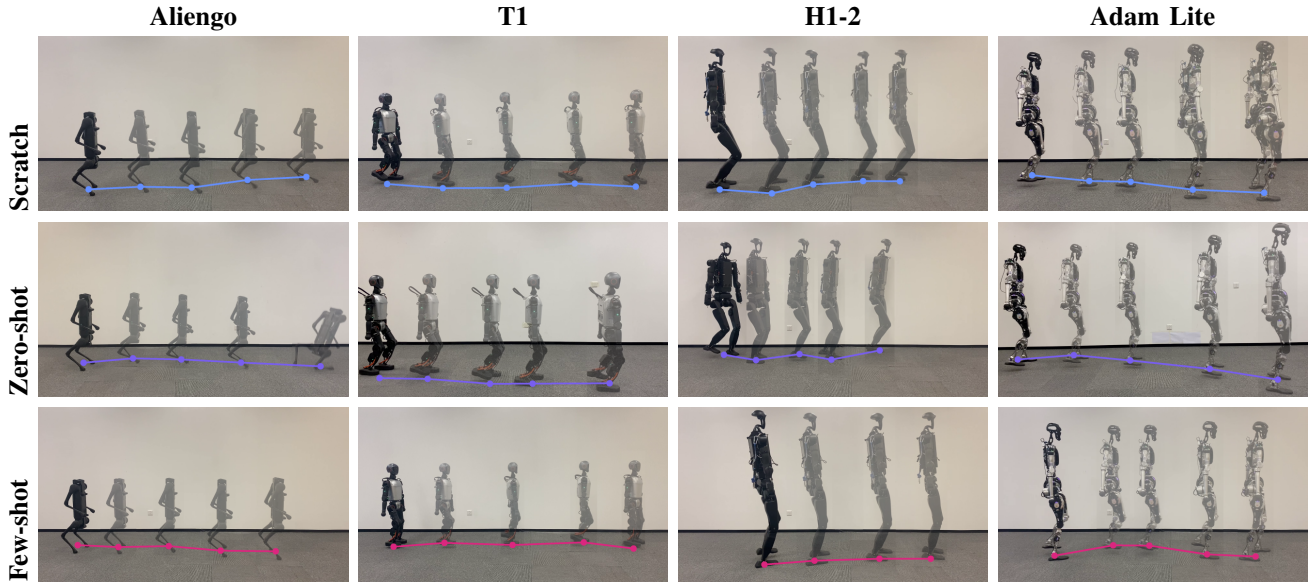

Fig. 7: **Snapshots from real-world deployment of learned policies.** Columns correspond to target robots, and rows denote training regimes: scratch training (top), zero-shot transfer (middle), and few-shot adaptation (bottom). While the pretrained policy demonstrates moderate transferability to unseen robots, it often exhibits instability and velocity drift (highlighted by the line near the robot's foot). Fine-tuning mitigates these issues, resulting in more stable and consistent control.

C. Real-world performance

We evaluate the qualitative performance of the fine-tuned policies in the real world on multiple hardware platforms. Inference runs at 50 Hz on each robot's onboard compute. Snapshots of recorded deployments are shown in Figure 7. The pretrained base policy demonstrates moderate adaptability to several unseen robots but is prone to jittery motion, velocity drift, and instabilities at high velocity commands. Fine-tuning the policy for the target robot in only a few simulation epochs enables more stable locomotion and responsive control.

VI. Conclusion

We propose a simple yet effective pretraining method for humanoid locomotion by unifying control semantics and populating the training set with diverse embodiments that feature randomized physical parameters and action masks. This approach enables the policy to generalize across a wide range of morphologies and control interfaces.

Compared to single-robot training, the resulting pretrained policy demonstrates significantly stronger transferability to unseen robots, allowing for few-shot adaptation to novel embodiments without retraining from scratch.

However, our current method relies on choosing a fixed set of training embodiments and randomization ranges, which may not fully capture the diversity of real-world robots. Future work includes exploring curriculum-based embodiment selection and randomization shaping to enable better generalization for whole-body control.

ACKNOWLEDGMENT

The SJTU team is supported by National Natural Science Foundation of China (62322603), Shanghai Municipal Science and Technology Major Project (2025SHZDZX025D08) and Shanghai Artificial Intelligence Laboratory.

References

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

[2] A. Kumar, Z. Fu, D. Pathak, and J. Malik, "Rma: Rapid motor adaptation for legged robots," *arXiv preprint arXiv:2107.04034*, 2021.

[3] T. Miki, J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning robust perceptive locomotion for quadrupedal robots in the wild," *Science robotics*, vol. 7, no. 62, p. eabk2822, 2022.

[4] Z. Fu, X. Cheng, and D. Pathak, "Deep whole-body control: Learning a unified policy for manipulation and locomotion," in *Conference on Robot Learning (CoRL)*, 2022.

[5] F. Jenelten, J. He, F. Farshidian, and M. Hutter, "Dtc: Deep tracking control–a unifying approach to model-based planning and reinforcement-learning for versatile and robust locomotion," *arXiv preprint arXiv:2309.15462*, 2023.

[6] J. Wu, G. Xin, C. Qi, and Y. Xue, "Learning robust and agile legged locomotion using adversarial motion priors," *IEEE Robotics and Automation Letters*, 2023.

[7] Z. Fu, A. Kumar, J. Malik, and D. Pathak, "Minimizing energy consumption leads to the emergence of gaits in legged robots," *arXiv preprint arXiv:2111.01674*, 2021.

[8] T. He, C. Zhang, W. Xiao, G. He, C. Liu, and G. Shi, "Agile but safe: Learning collision-free high-speed legged locomotion," in *Robotics: Science and Systems (RSS)*, 2024.

[9] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Science robotics*, vol. 5, no. 47, p. eabc5986, 2020.

[10] L. Smith, I. Kostrikov, and S. Levine, "A walk in the park: Learning to walk in 20 minutes with model-free reinforcement learning," *arXiv preprint arXiv:2208.07860*, 2022.

[11] D. Hoeller, N. Rudin, D. Sako, and M. Hutter, "Anymal parkour: Learning agile navigation for quadrupedal robots," *Science Robotics*, vol. 9, no. 88, p. eadi7566, 2024.

[12] J. Lee, M. Bjelonic, A. Reske, L. Wellhausen, T. Miki, and M. Hutter, "Learning robust autonomous navigation and locomotion for wheeled-legged robots," *Science Robotics*, vol. 9, no. 89, p. eadi9641, 2024.

[13] X. Gu, Y.-J. Wang, X. Zhu, C. Shi, Y. Guo, Y. Liu, and J. Chen, "Advancing humanoid locomotion: Mastering challenging terrains with denoising world model learning," *arXiv preprint arXiv:2408.14472*, 2024.

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

[15] H. Wang, Z. Wang, J. Ren, Q. Ben, T. Huang, W. Zhang, and J. Pang, "Beamdojo: Learning agile humanoid locomotion on sparse footholds," *arXiv preprint arXiv:2502.10363*, 2025.

[16] T. Huang, J. Ren, H. Wang, Z. Wang, Q. Ben, M. Wen, X. Chen, J. Li, and J. Pang, "Learning humanoid standing-up control across diverse postures," *arXiv preprint arXiv:2502.08378*, 2025.

[17] H. Lai, W. Zhang, X. He, C. Yu, Z. Tian, Y. Yu, and J. Wang, "Sim-to-real transfer for quadrupedal locomotion via terrain transformer," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. London, United Kingdom: IEEE, May 2023, p. 5141–5147. [Online]. Available: https://ieeexplore.ieee.org/document/10160497/

[18] P. Wu, A. Escontrela, D. Hafner, P. Abbeel, and K. Goldberg, "Daydreamer: World models for physical robot learning," in *Conference on robot learning*. PMLR, 2023, pp. 2226–2240.

[19] G. B. Margolis and P. Agrawal, "Walk these ways: Tuning robot control for generalization with multiplicity of behavior," in *Conference on Robot Learning*. PMLR, 2023, pp. 22–31.

[20] K. Caluwaerts, A. Iscen, J. C. Kew, W. Yu, T. Zhang, D. Freeman, K.-H. Lee, L. Lee, S. Saliceti, V. Zhuang *et al.*, "Barkour: Benchmarking animal-level agility with quadruped robots," *arXiv preprint arXiv:2305.14654*, 2023.

[21] X. Cheng, K. Shi, A. Agarwal, and D. Pathak, "Extreme parkour with legged robots," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 11 443–11 450.

[22] T. He, W. Xiao, T. Lin, Z. Luo, Z. Xu, Z. Jiang, J. Kautz, C. Liu, G. Shi, X. Wang *et al.*, "Hover: Versatile neural whole-body controller for humanoid robots," *arXiv preprint arXiv:2410.21229*, 2024.

[23] X. Cheng, Y. Ji, J. Chen, R. Yang, G. Yang, and X. Wang, "Expressive whole-body control for humanoid robots," *arXiv preprint arXiv:2402.16796*, 2024.

[24] M. Ji, X. Peng, F. Liu, J. Li, G. Yang, X. Cheng, and X. Wang, "Exbody2: Advanced expressive humanoid whole-body control," *arXiv preprint arXiv:2412.13196*, 2024.

[25] K. Yin, W. Zeng, K. Fan, Z. Wang, Q. Zhang, Z. Tian, J. Wang, J. Pang, and W. Zhang, "Unitracker: Learning universal whole-body motion tracker for humanoid robots," 2025. [Online]. Available: https://arxiv.org/abs/2507.07356

[26] T. He, Z. Luo, W. Xiao, C. Zhang, K. Kitani, C. Liu, and G. Shi, "Learning human-to-humanoid real-time whole-body teleoperation," *arXiv preprint arXiv:2403.04436*, 2024.

[27] T. He, Z. Luo, X. He, W. Xiao, C. Zhang, W. Zhang, K. Kitani, C. Liu, and G. Shi, "Omnih2o: Universal and dexterous human-to-humanoid whole-body teleoperation and learning," *arXiv preprint arXiv:2406.08858*, 2024.

[28] W. Huang, I. Mordatch, and D. Pathak, "One policy to control them all: Shared modular policies for agent-agnostic control," in *International Conference on Machine Learning*. PMLR, 2020, pp. 4455–4464.

[29] J. Whitman, M. Travers, and H. Choset, "Learning modular robot control policies," *IEEE Transactions on Robotics*, 2023.

[30] C. Yu, W. Zhang, H. Lai, Z. Tian, L. Kneip, and J. Wang, "Multi-embodiment legged robot control as a sequence modeling problem," *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7250–7257, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:254854044

[31] N. Bohlinger, G. Czechmanowski, M. Krupka, P. Kicki, K. Walas, J. Peters, and D. Tateo, "One policy to run them all: an end-to-end learning approach to multi-embodiment locomotion," *arXiv preprint arXiv:2409.06366*, 2024.

[32] R. Doshi, H. R. Walke, O. Mees, S. Dasari, and S. Levine, "Scaling cross-embodied learning: One policy for manipulation, navigation, locomotion and aviation," in *8th Annual Conference on Robot Learning*, 2024.

[33] S. Yang, Z. Fu, Z. Cao, G. Junde, P. Wensing, W. Zhang, and H. Chen, "Multi-loco: Unifying multi-embodiment legged locomotion via reinforcement learning augmented diffusion," 2025. [Online]. Available: https://arxiv.org/abs/2506.11470

[34] A. Patel and S. Song, "Get-zero: Graph embodiment transformer for zero-shot embodiment generalization," *ArXiv*, vol. abs/2407.15002, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:271328287

[35] G. Feng, H. Zhang, Z. Li, X. B. Peng, B. Basireddy, L. Yue, Z. Song, L. Yang, Y. Liu, K. Sreenath *et al.*, "Genloco: Generalized locomotion controllers for quadrupedal robots," in *Conference on Robot Learning*. PMLR, 2023, pp. 1893–1903.

[36] B. Ai, L. Dai, N. Bohlinger, D. Li, T. Mu, Z. Wu, K. Fay, H. I. Christensen, J. Peters, and H. Su, "Towards embodiment scaling laws in robot locomotion," 2025. [Online]. Available: https://arxiv.org/abs/2505.05753

[37] X. B. Peng, E. Coumans, T. Zhang, T.-W. Lee, J. Tan, and S. Levine, "Learning agile robotic locomotion skills by imitating animals," *arXiv preprint arXiv:2004.00784*, 2020.

[38] Z. Xie, X. Da, M. van de Panne, B. Babich, and A. Garg, "Dynamics randomization revisited: A case study for quadrupedal locomotion," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 4955–4961.

[39] I. M. A. Nahrendra, B. Yu, and H. Myung, "Dreamwaq: Learning robust quadrupedal locomotion with implicit terrain imagination via deep reinforcement learning," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 5078–5084.

[40] T. Yu, D. Quillen, Z. He, R. Julian, K. Hausman, C. Finn, and S. Levine, "Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning," in *Conference on robot learning*. PMLR, 2020, pp. 1094–1100.

[41] N. Bohlinger, G. Czechmanowski, M. Krupka, P. Kicki, K. Walas, J. Peters, and D. Tateo, "One policy to run them all: an end-to-end learning approach to multi-embodiment locomotion," *Conference on Robot Learning*, 2024.

[42] S. Choi, G. Ji, J. Park, H. Kim, J. Mun, J. H. Lee, and J. Hwangbo, "Learning quadrupedal locomotion on deformable terrain," *Science Robotics*, vol. 8, no. 74, p. eade2256, 2023.

[43] E. Valassakis, Z. Ding, and E. Johns, "Crossing the gap: A deep dive into zero-shot sim-to-real transfer for dynamics," *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5372–5379, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:221140175

[44] L. Campanaro, S. Gangapurwala, W. Merkt, and I. Havoutis, "Learning and deploying robust locomotion policies with minimal dynamics randomization," *arXiv preprint arXiv:2209.12878*, 2022.

[45] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 23–30.

[46] Unitree, "Unitree robotics," https://www.unitree.com/, 2022.

[47] PNDbotics, "Pndbotics website," https://pndbotics.com/humanoid, 2025.

[48] FourierIntelligence, "Fourier," https://github.com/FFTAI, 2022.

[49] BoosterRobotics, "Booster robotics official website," https://www.boosterobotics.com/robots/, 2025.

[50] EngineAI, "Engineai-home," https://www.engineai.com.cn/product_fore, 2025.

[51] OpenLoong, "loongopen/openloong-hardware," https://github.com/loongOpen/OpenLoong-Hardware, 2025.

[52] LejuRobotics, "kuavo-ros-opensource," https://gitee.com/leju-robot/kuavo-ros-opensource, 2025.

[53] Dobot, "Dobot atom," https://www.dobot-robots.com/products/humanoid-robots/atom.html, 2025.

[54] L. van der Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. 86, pp. 2579–2605, 2008. [Online]. Available: http://jmlr.org/papers/v9/vandermaaten08a.html

[55] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," *CoRR*, vol. abs/1511.07289, 2015.

[56] E. Xiao, Y. Dong, J. Lam, and P. Lu, "Learning stable bipedal locomotion skills for quadrupedal robots on challenging terrains with automatic fall recovery," *npj Robotics*, vol. 3, no. 1, p. 22, 2025.

[57] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, and G. State, "Isaac gym: High performance gpu-based physics simulation for robot learning," *ArXiv*, vol. abs/2108.10470, 2021.