PUMA: Perception-driven Unified Foothold Prior for Mobility Augmented Quadruped Parkour

Liang Wang[1,2], Kanzhong Yao[2], Yang Liu[2], Weikai Qin[2], Jun Wu[1], Zhe Sun[*2] and Qiuguo Zhu[*1]

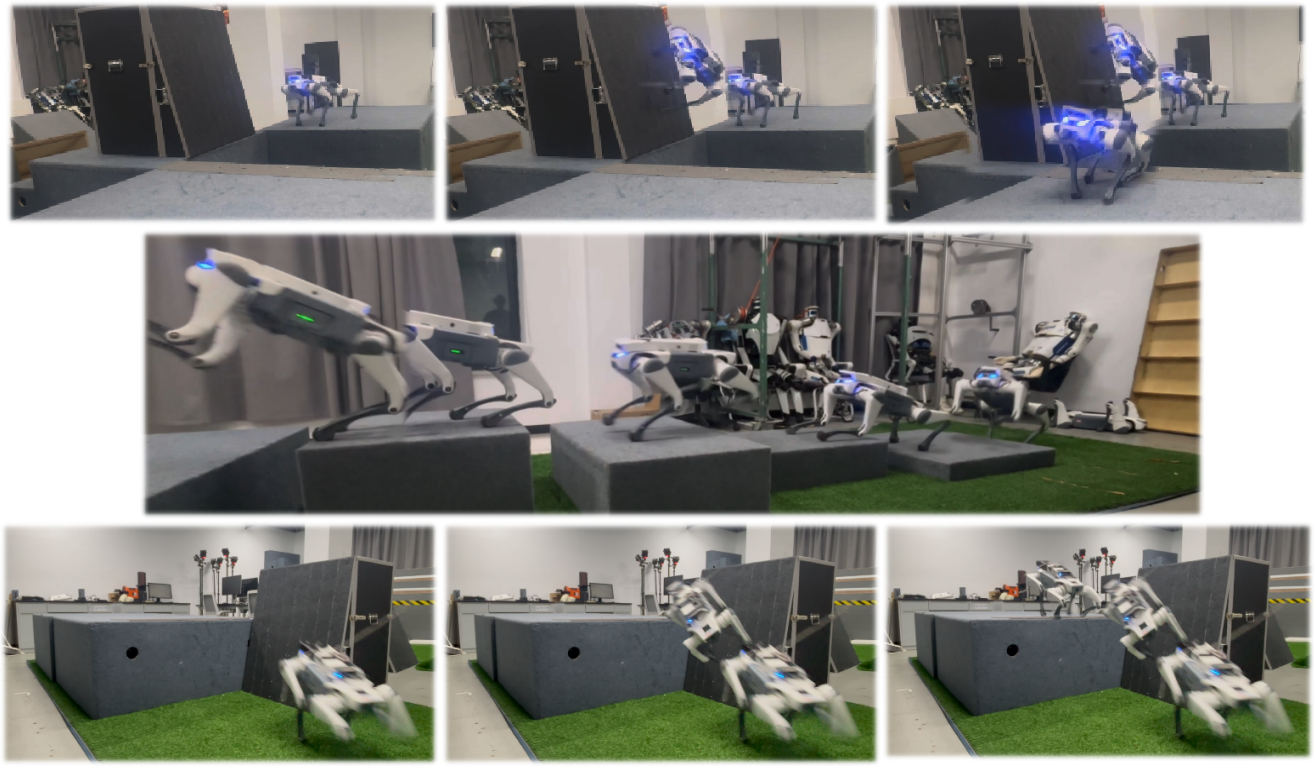


Fig. 1: **PUMA** enables quadruped robots to fuse proprioception with visual perception to estimate adaptive footholds for traversing complex discrete terrains. **Top Row:** The robot twists its posture to forcefully kick off the inclined wall, propelling itself across a wide gap. **Middle Row:** The robot sequentially traverses uneven stepping stones. **Bottom Row:** The robot leverages an inclined wall to surmount a high platform.

***Abstract*— Parkour tasks for quadrupeds have emerged as a promising benchmark for agile locomotion. While human athletes can effectively perceive environmental characteristics to select appropriate footholds for obstacle traversal, endowing legged robots with similar perceptual reasoning remains a significant challenge. Existing methods often rely on hierarchical controllers that follow pre-computed footholds, thereby constraining the robot's real-time adaptability and the exploratory potential of reinforcement learning. To overcome these challenges, we present PUMA, an end-to-end learning framework that integrates visual perception and foothold priors into a single-stage training process. This approach leverages terrain features to estimate egocentric polar foothold priors, composed of relative distance and heading, guiding the robot in active posture adaptation for parkour tasks. Extensive experiments conducted in simulation and real-world environments across various discrete complex terrains, demonstrate PUMA's exceptional agility and robustness in challenging scenarios.**

[1]The authors are with Institute of Cyber-Systems and Control,Zhejiang University, 310027, China 3210102182@zju.edu.cn.

[2]Institute of Artificial Intelligence (TeleAI), China Telecom.

[*]Corresponding authors: Qiuguo Zhu (qgzhu@zju.edu.cn) and Zhe Sun (sunzhe@nwpu.edu.cn).

***Index Terms*: foothold prior, visual perception, reinforcement learning.**

I. INTRODUCTION

In recent years, quadruped robots have demonstrated remarkable athletic performance: a low-cost quadruped robot can leap over long gaps, climb over high obstacles, and traverse complex discrete terrain composed of stepping stones and sloped platforms [1]–[6]. To navigate complex terrains in parkour tasks, many studies leverage exteroceptive sensors (e.g., cameras, LiDAR) to acquire terrain information. The sensory data is typically fused with proprioception to train dynamic locomotion policies via learning-based methods. Approaches include both decoupled hierarchical frameworks separating control and perception modules [7], [8], and end-to-end visual-aided policies [9], [10], establishing a strong foundation for agile legged locomotion.

However, even with the integration of exteroceptive sensing, robots still struggle to fully comprehend or exploit terrain features. Human parkour athletes can leverage en-

vironmental features to extend locomotive potential beyond inherent physical limits, exemplified by kicking off a wall to gain extra height and reach otherwise inaccessible elevations. This ability to strategically leverage terrain features for accomplishing locomotion tasks beyond actuator constraints remains a critical challenge in current legged robots.

To enable robots to traverse complex terrains with diverse geometric features, recent studies have concentrated on foot placement planning in locomotion. These approaches commonly adopt a hierarchical framework, where high-level trajectory optimization or learning-based methods are used to plan desired foothold locations based on terrain information. The planned footholds subsequently serve as execution objectives for the low-level tracking policy. Such methods have demonstrated robust locomotion capabilities across complex discrete terrains and deformable terrains [1], [11], [12]. However, the strict foothold tracking inherently constrains the robot's movement repertoire and impedes exploratory policy learning. Furthermore, the reliance on precise foothold tracking requires high-fidelity perception, imposing stringent sensing requirements that, in highly dynamic and contact-rich parkour tasks, significantly exacerbate the sim-to-real gap and hinder real-world deployment.

In this paper, we present PUMA, a **P**erception-driven **U**nified foothold prior framework for **M**obility **A**ugmented quadruped parkour. Unlike methods mentioned before, our method does not explicitly enforce foothold tracking. Instead, we employ egocentric polar footholds as motion priors for velocity tracking, which decomposes the explicit coordinates into relative distance and heading. By fusing depth perception with proprioception, PUMA estimates the polar footholds and feeds it directly into the actor network. To ensure stable convergence within this unified single-stage framework, we employ Probability Annealing Selection (PAS) method to gradually transition from ground-truth to predicted footholds during training. Extensive experiments on both simulated and physical robots equipped with onboard depth cameras demonstrate that PUMA enables robust traversal of challenging discrete terrains, such as uneven stepping stones, and allows the robot to clear wide gaps by strategically exploiting inclined walls, as shown in Fig. 1. The main contributions of this paper are as follows:

- A unified one-stage training framework that incorporates predicted footholds for robust locomotion over discrete terrains.
- A learned egocentric polar foothold prior that fuses proprioceptive and exteroceptive perception to guide velocity tracking without explicit foothold tracking.
- Extensive simulation and real-world experiments demonstrating robust sim-to-real transfer and agile locomotion using onboard depth sensing.

II. RELATED WORK

A. Robot Parkour

Quadruped robots have demonstrated agile motor skills with model-based methods [13], [14]. However, these approaches rely on precise environment modeling and exhibit limitations in complex, dynamically changing scenarios. In recent years, Reinforcement Learning (RL) has made significant progress in parkour tasks. Hoeller et al. [7] employed a hierarchical framework with separate locomotion control and perception modules, enabling navigation across highly complex terrains. Cheng et al. [3] and Zhuang et al. [4] utilized depth map information through a two-stage teacher-student structure to estimate privileged information and distill parkour locomotion policies. Luo et al. [2] further integrated the dual-stage approach into a single-stage framework, leveraging an asymmetric actor-critic network to learn parkour policies capable of implicitly imagining privileged observations. Despite these advances, relying solely on RL exploration in the absence of prior knowledge limits the strategic exploitation of terrain features.

B. Locomotion with Priors

To enhance the efficiency of RL exploration in locomotion tasks, incorporating motion priors into learning frameworks has emerged as a key approach. A common paradigm is to leverage reference trajectories in conjunction with imitation learning, enabling quadruped robots to acquire agile and dynamic behaviors [15], [16]. These approaches, however, rely heavily on high-quality expert demonstrations, which are typically obtained through motion capture systems or manually curated datasets [17], [18]. In practice, collecting such data is expensive and task-specific, and often requires additional processing or refinement, such as trajectory filtering, retargeting, or optimization-based smoothing [19], [20]. This dependence limits scalability and reduces adaptability to new environments or tasks.

C. Foothold-based Locomotion

More recent approaches directly condition policy objectives on explicit footholds as task-specific targets. For instance, Kim et al. [1] leverage a hierarchical framework combining sampling-based foothold planning and a learning-based tracker module to achieve high-speed navigation over discrete terrains. Coholich et al. [12] utilizes a high-level policy optimizes foothold targets using the low-level policy's value function without additional training. Jenelten et al. [11] utilize trajectory optimization to generate optimized footholds combined with RL for robust tracking. These methods demonstrate that footholds can effectively enhance robot locomotion performance across complex terrains. However, such decoupled frameworks generally necessitate high-fidelity terrain perception to execute accurate foothold tracking, which complicates real-world deployment. To address this challenge, we introduce egocentric foothold prior within a single-stage training framework to achieve robust terrain traversal.

III. METHOD

Our goal is to train an end-to-end velocity-tracking locomotion policy that integrates geometric features from an onboard depth camera with proprioception to accomplish parkour tasks over complex terrains. To address partial

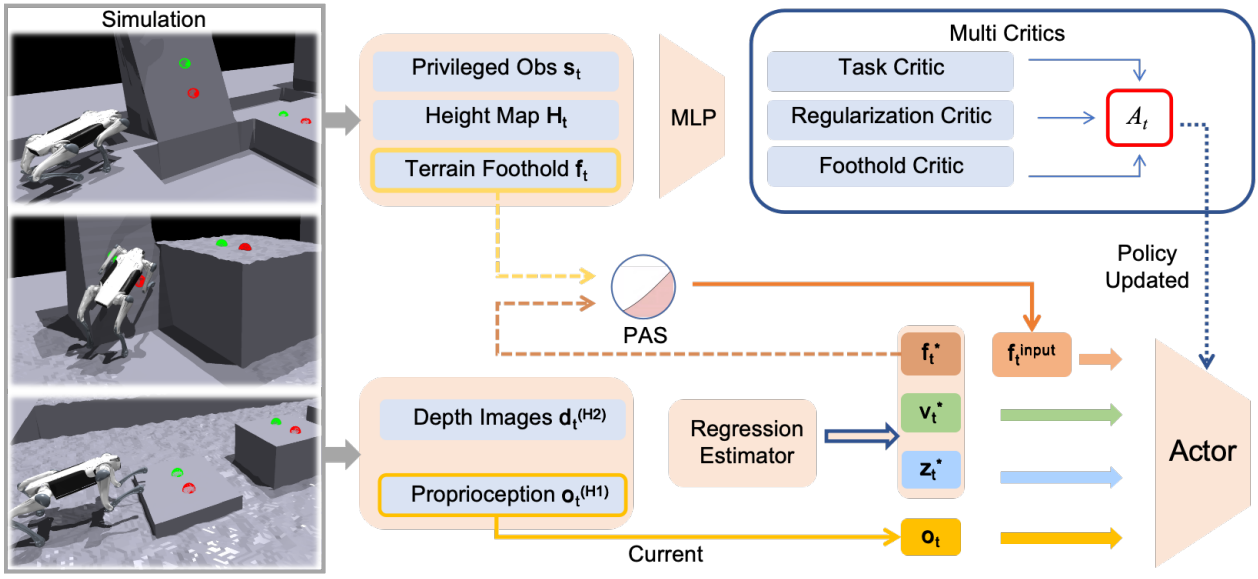


Fig. 2: Overview of PUMA training framework. A velocity-tracking locomotion policy takes proprioception and depth images as input to predict egocentric foothold priors, base velocity, and latent terrain features. These representations are then concatenated with the current observations and fed into the policy network. Multiple critic networks are trained on distinct reward components to cooperatively optimize the policy. During training, a PAS strategy gradually replaces ground-truth footholds with predicted ones. The entire process is conducted in a single stage, with all networks optimized simultaneously.

observability, we adopt an asymmetric actor-critic architecture [21], [22], as illustrated in Fig. 2. All modules in the training pipeline are optimized concurrently, trained from scratch without relying on pre-trained components.

A. Egocentric Foothold Prior Design

To mitigate inefficient exploration in high-dimensional action spaces, we introduce footholds in an egocentric manner as motion priors, rather than enforcing them as explicit tracking targets.

We denote the sequence of candidate footholds as $\{\mathbf{p}_i\}_{i=0}^{N}$, where each $\mathbf{p}_i \in \mathbb{R}^3$ represents the Cartesian coordinate in the world frame. These candidates are sampled at regular 1 m intervals along the robot's commanded direction, covering both flat terrain and the centers of exploitable inclined walls. Since the footholds are used as guidance priors rather than explicit tracking targets, complex kinematic feasibility checks are unnecessary. Instead, we focus on safety by filtering out points situated too close to hazardous edges. The final valid foothold set is defined as:

$$\mathcal{P} = \{\mathbf{p}_i \mid d_{\text{edge}}(\mathbf{p}_i) > d_{\text{safe}},\ i \in \{0, \dots, N\}\} \tag{1}$$

where $d_{\text{edge}}(\mathbf{p}_i)$ denotes the Euclidean distance from $\mathbf{p}_i$ to the nearest terrain edge, and d_{safe} is the safety threshold.

To avoid excessive foot guidance biasing the policy learning direction, we focus exclusively on the front feet. The coordinate $\mathbf{p}_t$ from the generated sequence is transformed into a polar representation, forming the proposed egocentric foothold prior. Formally, this feature vector $\mathbf{f}_t$ at time step t is formulated as:

$$\mathbf{f}_t = \left\{ d_t^{(L)},\ d_t^{(R)},\ \psi_t,\ \psi_{t+1} \right\}$$

Here, $d_t^{(L)}$ and $d_t^{(R)}$ denote the Euclidean distances from the robot's left and right forefeet to the current expected foothold $\mathbf{p}_t$. ψ_t and ψ_{t+1} represent the heading error between the robot's current orientation and the target direction toward $\mathbf{p}_t$ and $\mathbf{p}_{t+1}$, respectively.

B. Training Pipeline

Distinct from existing works [11], [12] that often adopt a hierarchical framework to separately handle foothold generation and locomotion, we employ a unified single-stage approach based on an asymmetric actor-critic architecture [21], [22]. Within this framework, the policy leverages the proposed egocentric foothold prior to directly learn parkour skills on complex terrains. The policy network is optimized using Proximal Policy Optimization (PPO).

1) Network Inputs: The policy network receives proprioceptive history $\mathbf{o}_t^{(H1)}$ and depth image buffer $\mathbf{d}_t^{(H2)}$ as input. Each proprioceptive observation $\mathbf{o}_t \in \mathbb{R}^{45}$ comprises the angular velocity $\boldsymbol{\omega}_t \in \mathbb{R}^3$, the gravity vector $\mathbf{g}_t \in \mathbb{R}^3$, the command $\mathbf{c}_t \in \mathbb{R}^3$, the joint positions $\boldsymbol{\theta}_t \in \mathbb{R}^{12}$, the joint velocities $\dot{\boldsymbol{\theta}}_t \in \mathbb{R}^{12}$, and the previous joint target positions $\mathbf{a}_{t-1} \in \mathbb{R}^{12}$.

The critic network, having access to privileged information, receives the privileged observation $\mathbf{s}_t$, surrounding heightfield $\mathbf{H}_t$ and the ground-truth egocentric foothold prior $\mathbf{f}_t$ as input. The privileged observation $\mathbf{s}_t$ is defined as:

$$\mathbf{s}_t = \left[\mathbf{o}_t,\ \mathbf{v}_t,\ \mathbf{p}_t,\ \mathbf{p}_{t+1}, \mathbf{x}_t\right]^T$$

where $\mathbf{o}_t$ denotes the current proprioceptive observation, $\mathbf{v}_t \in \mathbb{R}^3$ is the base velocity. The terms $\mathbf{p}_t, \mathbf{p}_{t+1}$ represent the Cartesian coordinates of the current and next expected footholds, while $\mathbf{x}_t \in \mathbb{R}^6$ denotes the Cartesian positions of the robot's forefeet.

2) Regression estimator: Building upon the implicit-explicit estimator proposed by Luo et al. [2], we extend this architecture to jointly infer the robot's internal states, environmental features, and the egocentric foothold prior from partial observations.

Specifically, the depth image buffer $\mathbf{d}_t^{(H2)}$ is first encoded via a Convolutional Neural Network (CNN). The extracted visual features are then concatenated with the proprioceptive history $\mathbf{o}_t^{(H1)}$, forming a token sequence which is processed through a self-attention mechanism. The resulting sequence is fed into a Gated Recurrent Unit (GRU) network for temporal modeling, followed by several separate Multi-Layer Perceptron (MLP) heads to regress the estimated egocentric foothold prior $\hat{\mathbf{f}}_t$, the base velocity $\hat{\mathbf{v}}_t$, and the environment latent $\hat{\mathbf{z}}_t \in \mathbb{R}^{64}$.

3) PAS in Foothold Prior: As the actor network receives the estimated foothold $\hat{\mathbf{f}}_t$ from the regression estimator, unreliable priors in the early stage of training can destabilize the policy's exploration, significantly impeding the learning process. To overcome the challenge, we apply the PAS method [23] to the foothold input, where the actor probabilistically receives either the ground-truth foothold or the estimator's prediction. Crucially, the probability of utilizing the ground-truth value gradually decreases as training progresses. At each training step, we sample a uniform random variable $u_t \sim \mathcal{U}(0,1)$ to decide whether to use the ground-truth foothold $\mathbf{f}_t$ or the estimator output $\hat{\mathbf{f}}_t$, based on the annealed probability p_t. The annealing schedule is defined as:

$$f_t^{\text{input}} = \begin{cases} \hat{f}_t, & u_t < p_t, \\ f_t, & u_t \geq p_t, \end{cases} \quad u_t \sim \mathcal{U}(0,1) \tag{2}$$

$$p_t = 1 - \cos\left(\frac{\pi t}{2T}\right) \tag{3}$$

where f_t^{input} denotes the input fed to the actor, t is the current training iteration, and T is the total number of annealing steps.

C. Rewards and Multi Critics

We design a composite reward structure based on the proposed egocentric polar foothold prior, incorporating heading alignment, dense distance tracking, and sparse arrival components. Beyond providing directional guidance, the reference footholds for the forefeet can implicitly encourage the robot to adapt its body posture to establish effective contact with the terrain. The foothold reward design is formulated as follows:

$$r_t^{\text{yaw}} = w_{\text{y}} \cdot \exp\left(-|\psi_t|\right) \tag{4}$$

$$r_t^{\text{dense}} = w_{\text{d}} \cdot \exp\left(-\left(d_t^{(L)} + d_t^{(R)}\right)\right) \tag{5}$$

$$r_t^{\text{sparse}} = w_{\text{s}} \cdot \mathbb{I}\left(d_t^{(L)} < \epsilon \wedge d_t^{(R)} < \epsilon\right) \tag{6}$$

Here, w_y, w_d and w_s denote the weights for yaw, dense and sparse rewards respectively. A fixed sparse reward is granted only when the distances of both the robot's left and right forefeet to their corresponding expected footholds simultaneously fall within the threshold ϵ.

TABLE I: Reward Functions and Weights

Name	Equation	Weight
Tracking Group		**3**
Lin. vel tracking	$\exp\{-4\|\mathbf{v}_{xy}^{\text{cmd}} - \mathbf{v}_{xy}\|^2\}$	1
Ang. vel tracking	$\exp\{-4\|\omega_z^{\text{cmd}} - \omega_z\|^2\}$	0.5
Foothold Group		**1.5**
Dense Foothold	$\exp\left(-(d_t^{(L)} + d_t^{(R)})\right)$	1
Sparse Foothold	$\mathbb{I}\left(d_t^{(L)} < \epsilon \wedge d_t^{(R)} < \epsilon\right)$	1
Yaw Foothold	$\exp\left(-\lvert\psi_t\rvert\right)$	1
Regularization Group		**1.0**
Linear velocity (z)	v_z^2	-1.0
Angular velocity (xy)	$\|\boldsymbol{\omega}_{xy}\|^2$	-0.05
Orientation	$\|\mathbf{g}_{xy}\|^2$	-1.0
Joint accelerations	$\|\ddot{\boldsymbol{\theta}}\|^2$	-2.5e-7
Joint power	$\sum \lvert\tau_j \dot{\theta}_j\rvert$	-2.0e-5
Collision	n_{col}	-10.0
Action rate	$\|\mathbf{a}_t - \mathbf{a}_{t-1}\|^2$	-0.01
Smoothness	$\|\mathbf{a}_t - 2\mathbf{a}_{t-1} + \mathbf{a}_{t-2}\|^2$	-0.01

To enhance the estimation of expected returns from the mixture of sparse and dense rewards [24], we categorize the rewards into different groups based on their types and adopt multi-critic approach [25]–[27] to independently estimate the return for each reward group:

$$r_t = w_{\text{task}} \cdot r_t^{\text{task}} + w_{\text{foothold}} \cdot r_t^{\text{foothold}} + w_{\text{style}} \cdot r_t^{\text{style}} \tag{7}$$

Here, r_t^{task}, r_t^{foothold}, and r_t^{style} represent the task, foothold, and regularization reward. The complete reward design is summarized in Table I. Each critic network V_{ϕ_i} is optimized independently for its corresponding group with temporal difference loss :

$$L(\phi_i) = \hat{E}_t\left[\left\| r_{i,t} + \gamma V_{\phi_i}(s_{t+1}) - \tilde{V}_{\phi_i}(s_t) \right\|^2\right] \tag{8}$$

Here, i denotes the reward group index, $r_{i,t}$ is the scalar reward for group i at step t, and $\tilde{V}_{\phi_i}$ represents the target value network.

To accommodate multiple reward components, we adopt a Multi-Critic (Muc) architecture, where each critic estimates the advantage associated with a specific reward term. The individual advantages are then combined into a unified advantage used for policy optimization. Specifically, the weighted sum of advantages is normalized as:

$$\hat{A}_{\text{MuC}} = \frac{\sum_{i=0}^{n} w_i \cdot \hat{A}_i - \mu_{\text{MuC}}}{\sigma_{\text{MuC}}} \tag{9}$$

$$L(\theta) = \mathbb{E}\big[\min\big(\alpha_t(\theta)\hat{A},\ \text{clip}(\alpha_t(\theta), 1-\epsilon, 1+\epsilon)\,\hat{A}\big)\big] \tag{10}$$

where μ and σ denote the batch mean and standard deviation of the weighted sum, $\alpha_t(\theta)$ is the probability ratio and ϵ the clipping hyper-parameter.

D. Terrain and curriculum Design

As illustrated in Fig. 3, We design three types of discrete terrains and establish a corresponding curriculum in simulation.

TABLE II: Success Rate (SR) and Traverse Rate (TR) comparison in simulation. Each cell shows SR TR (%).

Method	Stepping Stone	Wall-assisted Gap		Surmounting	
	SR TR	SR TR	SR TR	SR TR	SR TR
inclination angle		60°	80°	60°	80°
Ours	**98.7 99.4**	**98.6 97.3**	**96.2 95.3**	**96.8 96.9**	**94.7 92.5**
(a) Ablation on foothold design					
w/o Foothold Prior	69.6 76.3	54.7 57.5	24.7 20.5	49.4 47.9	5.3 17.4
w/o Relative Distance	98.3 99.1	88.7 85.4	80.5 84.1	55.9 54.3	6.9 18.0
Explicit Cartesian Prior	91.6 90.1	93.5 91.9	86.4 85.2	90.9 88.3	78.1 81.9
Implicit Cartesian Prior	93.8 90.3	93.7 89.8	88.1 86.0	90.3 90.6	80.1 83.3
(b) Ablation on PAS iteration					
w/o PAS	98.1 98.3	96.2 96.6	93.1 94.5	94.8 97.3	93.2 94.5
(c) Ablation on critics					
w/o MuC	96.8 95.9	90.9 87.7	45.7 46.7	89.0 87.5	19.9 36.8
(d) Baselines					
Extreme Parkour	81.1 78.4	77.2 76.5	3.2 14.0	71.8 68.2	– 14.3
PIE	67.3 77.2	45.6 42.5	12.1 21.5	37.7 32.1	– 13.9

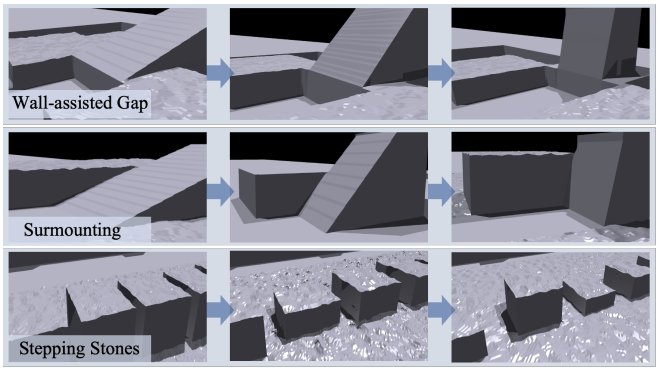


Fig. 3: Terrain difficulty gradually increasing from the left side towards the right. Notably, we introduce height variations to horizontal surfaces to simulate roughness, while the inclined walls remain smooth.

Wall-assisted Gap: This terrain features randomly spaced gaps, each edged with inclined stepping walls. As the curriculum progresses, both the gap width and wall inclination increase.

Surmounting: The terrain consists of elevated platforms with stepping walls attached to the leading edge. The platform height and wall inclinations are gradually increased throughout the curriculum.

Stepping Stones: The terrain comprises stepping stones of varying dimensions and heights. As the curriculum level advances, the horizontal spacing and vertical drops between the stones increase, while the length and width decrease.

IV. EXPERIMENTS

A. Experimental Setup

Simulation training. We train the 12-DoF DeepRobotics Lite3 robot using Isaac Gym across 2,048 environments on a single NVIDIA RTX 4090 GPU. The entire RL process is completed end-to-end without any pretraining. To maximize GPU memory efficiency during rollout, we implement a lightweight ray-tracing–based depth renderer built on NVIDIA Warp. To facilitate robust sim-to-real transfer, we further apply domain randomization on both observations and physical parameters, and introduce randomized delays in the visual input stream.

Real-world deployment. The trained policy is deployed on Lite3 robot onboard using an RK3588 computing unit, running inference at 50 Hz. The network outputs joint-level commands, which are translated into motor torques via PD control with gains $P = 20$ and $D = 0.5$. Depth observations are provided by the onboard Intel RealSense D435i camera, updating depth frames at 10 Hz.

We designed a set of experiments to demonstrate the effectiveness and advancement of our framework, as detailed below:

(a)Ablation on foothold design

- **w/o Foothold Prior:** The actor receives no foothold prior as input.
- **w/o Relative Distance:** The foothold prior includes only the yaw angle component, omitting the distance estimates.
- **Explicit Cartesian Prior:** The network directly regresses the explicit Cartesian coordinates of the desired foothold in the robot's frame.
- **Implicit Cartesian Prior:** The actor receives a compressed implicit foothold latent, which is reconstructed into Cartesian coordinates by an MLP decoder.

(b)Ablation on PAS iteration

- **w/o PAS:** Remove the PAS process during training.

(c)Ablation on critics

- **w/o MuC:** Employ a single critic network to estimate all reward functions.

(d)Baselines

- **PIE [2]:** A single-stage parkour framework that enables the robot to traverse elevated platforms and gaps.
- **Extreme Parkour [3]:** A two-stage parkour training framework that distills student policy from the teacher. We directly use the teacher policy here to bypass student limitations of imitation learning.

To ensure fair evaluations of the two baselines in our terrains, we retain their original reward functions while applying our terrain curriculum described in Section III-D.

B. Simulation Experiments

We conduct simulation experiments over the terrains described in Section III-D with a constant commanded robot velocity of 1.5 m/s. The locomotion performance of the different policies is measured using the following two metrics.

- **Success Rate (SR):** The probability of successfully crossing the entire terrain segment.
- **Traverse Rate (TR):** The ratio of the furthest distance reached in an attempt to the total length of the terrain.

We present the success rates and traverse rates of all methods on the three terrain classes in Table II, which are calculated from 1,000 trials in simulation. The experimental findings yield to the following insights:

1) Egocentric polar foothold prior extends the boundaries of parkour performance.

Our experimental results demonstrate that all methods incorporating foothold information outperform the baseline PIE and the ablation w/o foothold prior. Specifically, for the Extreme Parkour baseline, the heightfield input becomes excessively sparse and degrades on highly inclined terrains, failing to capture critical geometric details required for the actor to learn effective skills. PUMA, employing the proposed egocentric polar foothold prior, achieves superior and robust performance across all terrain types. We observe that the w/o relative distance variant performs adequately on Stepping Stones as yaw is sufficient to maintain heading direction between discrete terrains. However, its performance degrades on the other two terrain types, where the robot fails to learn the necessary body pose adjustment to make contact with highly inclined walls. In contrast, PUMA as well as the explicit and implicit Cartesian foothold priors performs effectively on steep terrains, as access to target foothold locations enables the robot to adjust its body posture and achieve stable foot contact. Notably, the two variants exhibit similar performance, since both aim to predict foothold coordinates.

To further investigate why the Cartesian-based methods underperform compared to PUMA, we evaluate the accuracy of foothold regression on the Wall-assisted Gap and Surmounting terrain. As shown in Table III, we quantitatively compare the Mean Square Error (MSE) between the predicted foothold value and the ground truth. Our method, which predicts an egocentric polar prior $\mathbf{f}_t$ composed of only four scalars, achieves significantly higher accuracy than the two Cartesian-based approaches. We attribute this superiority to the geometric nature of our representation: the polar format naturally decouples relative distance from heading direction. This decomposition simplifies the regression landscape compared to coupled Cartesian coordinates (x, y, z), enabling the network to learn a more precise and robust locomotion prior that facilitates posture adjustments for better terrain traversal and parkour maneuvers.

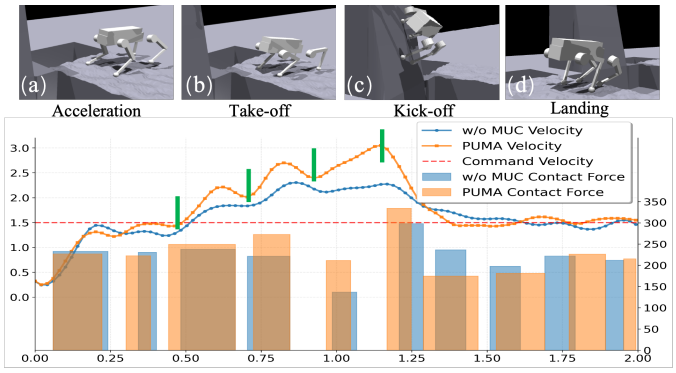


Fig. 4: Temporal evolution of body velocity and total contact force during a complete jump motion. The four motion phases are visualized in the top row and delimited by vertical green markers. The curves compare the body velocity of PUMA (orange) and the w/o MuC (blue) against the reference velocity (red dashed line), while the bottom bars represent the corresponding total contact forces.

TABLE III: Foothold Regression MSE(%).

Method	Terrain	
	Wall-assisted Gap	Surmounting
PUMA(ours)	6.33 ± 0.73	6.08 ± 0.29
Explicit Cartesian Prior	12.10 ± 1.20	11.17 ± 0.41
Implicit Cartesian Prior	9.30 ± 0.98	10.34 ± 1.15

2)Multi-critic design facilitates coordination between task and guidance rewards.

The success rate of w/o MuC drops significantly on highly inclined terrain. To better understand this behavior, we analyze the robot's velocity and contact forces during wall-assisted jumps across long gap, as shown in Figure 4. A complete jump motion can be divided into four phases: acceleration, take-off, kick-off, and landing. During the acceleration phase, the robot deviates from the commanded velocity, actively accelerating to build momentum. In the take-off phase, it executes a forceful jump. At this moment, the ground reaction force propels the robot forward while reorienting its body normal to the inclined surface. The robot gains a higher speed compared to the acceleration phase. In the kick-off phase, the robot forcefully pushes the feet against the wall, undergoing secondary acceleration to continue the jump. Finally, in the landing phase, the robot decelerates upon reaching horizontal ground, resuming tracking of the commanded velocity.

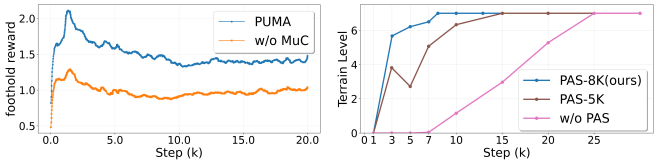
It can be observed that PUMA exerts greater force during the take-off and kick-off phases, achieving higher peak velocities. In contrast, the policy from w/o MuC generates insufficient thrust and fails to harness adequate reaction force from the inclined wall, thereby compromising its subsequent motion and leading to task failure. To accomplish high-difficulty parkour tasks in velocity tracking tasks, the robot must balance distinct objectives: it needs to temporarily violate velocity tracking constraints to satisfy foothold guidance and dynamic requirements. Hence, some sacrifice in velocity tracking performance is warranted. As shown in Fig. 6a,

Method	Success Rate	
	60°	80°
PUMA (Ours)	1.0	1.0
w/o Relative Distance	0.2	0.0
Explicit Cartesian Prior	0.6	0.2
Implicit Cartesian Prior	0.7	0.3
w/o MuC	0.6	0.0

Method	Success Rate
PUMA (Ours)	1.0
w/o Relative Distance	1.0
Explicit Cartesian Prior	0.5
Implicit Cartesian Prior	0.6
w/o MuC	0.7

Method	Success Rate	
	60°	80°
PUMA (Ours)	1.0	0.8
w/o Relative Distance	0.0	0.0
Explicit Cartesian Prior	0.4	0.1
Implicit Cartesian Prior	0.4	0.1
w/o MuC	0.2	0.0

Fig. 5: Real-world experimental results: (left) Wall-assisted Gap terrain, featuring stepping walls at 60° and 80° angles followed by a 1.2 m wide gap; (center) Stepping Stones terrain composed of discrete stones 0.5–0.8 m in length/width and 0–0.4 m in height variation; (right) Surmounting terrain with stepping walls at 60° and 80° angles leading to a 0.7 m high platform.



(a) Foothold Rewards (b) Learning Efficiency

Fig. 6: Training curves analysis. (a) PUMA obtains significantly higher rewards than the w/o MuC baseline in foothold group. (b) Comparison of learning efficiency with different annealing schedules, where 5K and 8K denote that the annealing process concludes at the corresponding training steps.

the w/o MuC method fails to effectively acquire foothold rewards, attributed to the inaccurate value estimation of foothold priors. In contrast, our PUMA framework demonstrates a superior capability in accurately estimating dense and sparse rewards across different categories, maintaining an optimal balance between the velocity task reward and the auxiliary foothold guidance.

3) PAS boosts learning in concurrent network training. Our results indicate that PAS method does not correlate with the locomotion performance. Through an additional ablation study on the annealing duration, we find that the efficacy of PAS is primarily manifested in a marked acceleration of policy convergence during training. The learning efficiency across different methods is illustrated in Fig. 6b. The training of the w/o PAS method progressed at a markedly slow pace, hardly learning an effective policy in the early stage. In comparison, the PAS-5K method and our approach exhibited comparable learning efficiency. However, the performance of PAS-5K degrades significantly during the late annealing phase. This decline likely stems from the premature conclusion of the annealing schedule, which thereby forces the actor to rely on the regression estimator before it has fully converged. The resulting inaccurate foothold priors act as noisy inputs, destabilizing the policy and leading to a decline in the curriculum level.

C. Real World Experiments

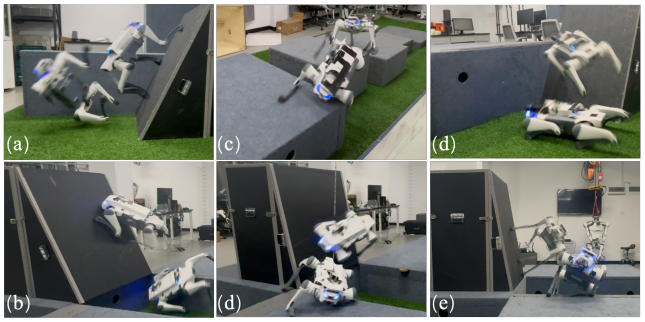


Fig. 7: Typical failure cases in the real-world experiment: (a)—(b) Correct yaw but failed roll adaptation leading to sliding down (a) or collision (b); (c) Misorientation leading to a missed step; (d) Foothold estimation error causing missed wall contact; (e) Insufficient jump height causing platform collision; (f) The robot modifies its body posture to make contact with the inclined wall but fails to generate effective propulsion, resulting in an insufficient jump distance.

We evaluate the zero-shot transfer performance of our proposed method and the ablations in real-world scenarios, as shown in Fig. 5. The evaluation is conducted across each terrain with 10 trials performed for each method.

The physical experiments demonstrate that our approach enables the robot to execute highly dynamic parkour tasks. Notably, the robot autonomously emerges a galloping gait and exhibits the capability to rapidly traverse discrete terrains. It effectively leverages stepping walls to gain kinetic energy, allowing it to leap across wide gaps and vault over high platforms.

Additionally, we analyze the failure cases encountered during the real-world experiments, with the results presented

in Fig 7. Due to multifaceted noise in the real-world environment, the foothold estimation accuracy of the explicit cartesian prior and implicit cartesian prior in foothold methods remains unstable, often leading to erroneous take-offs. These errors manifest primarily in incorrect body orientation and improper relative height during wall contact. While the w/o Relative Distance lacks the capability to adjust body posture for leveraging terrain features, it is noteworthy that it performs adequately on stepping-stone terrain. The w/o MuC method demonstrates poor real-world performance: although it can modify body posture, it consistently fails to establish effective force contact with the stepping wall, often resulting in false or grazing contact. In contrast, our proposed method successfully mitigates these issues, exhibiting remarkable stability and robustness.

V. CONCLUSION

In this work, we proposed PUMA, an end-to-end learning framework that enables quadruped robots to perceive terrain geometry from onboard visual sensing, infer egocentric foothold priors, and leverage them as motion guidance for robust locomotion over discrete and challenging terrains. By integrating foothold-aware perception with a unified learning framework, PUMA enables stable posture adaptation and dynamic traversal without relying on explicit foothold tracking or hierarchical planning.

Despite these promising results, several limitations remain. The current system relies primarily on geometric information from depth observations and does not explicitly reason about semantic or material properties of the environmental structures. In addition, the policy is trained and evaluated in static environments, and does not yet account for dynamic or deformable terrain interactions.

Future work will focus on incorporating semantic understanding and temporal reasoning into the foothold representation, as well as extending the framework to handle dynamic environments and more complex interaction scenarios. These directions are expected to further improve robustness, generalization, and real-world applicability of learning-based legged locomotion systems.

REFERENCES

[1] H. Kim, H. Oh, J. Park, Y. Kim, D. Youm, M. Jung, M. Lee, and J. Hwangbo, "High-speed control and navigation for quadrupedal robots on complex and discrete terrain," *Science Robotics*, vol. 10, no. 102, p. eads6192, 2025.

[2] S. Luo, S. Li, R. Yu, Z. Wang, J. Wu, and Q. Zhu, "Pie: Parkour with implicit-explicit learning framework for legged robots," *IEEE Robotics and Automation Letters*, 2024.

[3] 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.

[4] Z. Zhuang, Z. Fu, J. Wang, C. G. Atkeson, S. Schwertfeger, C. Finn, and H. Zhao, "Robot parkour learning," in *Conference on Robot Learning*. PMLR, 2023, pp. 73–92.

[5] J. He, C. Zhang, F. Jenelten, R. Grandia, M. Bächer, and M. Hutter, "Attention-based map encoding for learning generalized legged locomotion," *Science Robotics*, vol. 10, no. 105, p. eadv3604, 2025.

[6] R. Yu, Q. Wang, H. Li, Z. Jun, Z. Wang, J. Wu, and Q. Zhu, "Start: Traversing sparse footholds with terrain reconstruction," *IEEE Robotics and Automation Letters*, pp. 1–8, 2025.

[7] 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.

[8] Z. Fu, A. Kumar, A. Agarwal, H. Qi, J. Malik, and D. Pathak, "Coupling vision and proprioception for navigation of legged robots," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17 273–17 283.

[9] R. Yang, M. Zhang, N. Hansen, H. Xu, and X. Wang, "Learning vision-guided quadrupedal locomotion end-to-end with cross-modal transformers," in *Deep RL Workshop NeurIPS 2021*.

[10] A. Agarwal, A. Kumar, J. Malik, and D. Pathak, "Legged locomotion in challenging terrains using egocentric vision," in *Conference on robot learning*. PMLR, 2023, pp. 403–415.

[11] F. Jenelten, J. He, F. Farshidian, and M. Hutter, "Dtc: Deep tracking control," *Science Robotics*, vol. 9, no. 86, p. eadh5401, 2024.

[12] J. Coholich, M. A. Murtaza, S. Hutchinson, and Z. Kira, "Hierarchical reinforcement learning and value optimization for challenging quadruped locomotion," *arXiv preprint arXiv:2506.20036*, 2025.

[13] Q. Nguyen, M. J. Powell, B. Katz, J. Di Carlo, and S. Kim, "Optimized jumping on the mit cheetah 3 robot," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 7448–7454.

[14] C. Nguyen, L. Bao, and Q. Nguyen, "Continuous jumping for legged robots on stepping stones via trajectory optimization and model predictive control," in *2022 IEEE 61st Conference on Decision and Control (CDC)*. IEEE, 2022, pp. 93–99.

[15] X. B. Peng, P. Abbeel, S. Levine, and M. Van de Panne, "Deepmimic: Example-guided deep reinforcement learning of physics-based character skills," *ACM Transactions On Graphics (TOG)*, vol. 37, no. 4, pp. 1–14, 2018.

[16] X. B. Peng, Z. Ma, P. Abbeel, S. Levine, and A. Kanazawa, "Amp: Adversarial motion priors for stylized physics-based character control," *ACM Transactions on Graphics (ToG)*, vol. 40, no. 4, pp. 1–20, 2021.

[17] A. Escontrela, X. B. Peng, W. Yu, T. Zhang, A. Iscen, K. Goldberg, and P. Abbeel, "Adversarial motion priors make good substitutes for complex reward functions," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 25–32.

[18] J. Wu, G. Xin, C. Qi, and Y. Xue, "Learning robust and agile legged locomotion using adversarial motion priors," *IEEE Robotics and Automation Letters*, vol. 8, no. 8, pp. 4975–4982, 2023.

[19] T. Yoon, D. Kang, S. Kim, J. Cheng, M. Ahn, S. Coros, and S. Choi, "Spatio-temporal motion retargeting for quadruped robots," *IEEE Transactions on Robotics*, 2025.

[20] R. Watanabe, C. Li, and M. Hutter, "Dfm: Deep fourier mimic for expressive dance motion learning," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*, 2025, pp. 9644–9650.

[21] 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 2023*. Institute of Electrical and Electronics Engineers Inc., 2023, pp. 5078–5084.

[22] L. Pinto, M. Andrychowicz, P. Welinder, W. Zaremba, and P. Abbeel, "Asymmetric actor critic for image-based robot learning," in *Proceedings of Robotics: Science and Systems*, Pittsburgh, Pennsylvania, June 2018.

[23] S. Zhu, D. Li, L. Mou, Y. Liu, N. Xu, and H. Zhao, "Saro: Space-aware robot system for terrain crossing via vision-language model," pp. 14 820–14 827, 2025.

[24] F. Zargarbashi, J. Cheng, D. Kang, R. Sumner, and S. Coros, "Robotkeyframing: Learning locomotion with high-level objectives via mixture of dense and sparse rewards," in *Conference on Robot Learning*. PMLR, 2025, pp. 916–932.

[25] A. E. Vijayan, A. Cramariuc, M. Risiglione, C. Gehring, and M. Hutter, "Multi-critic learning for whole-body end-effector twist tracking," in *Conference on Robot Learning*. PMLR, 2025, pp. 1470–1485.

[26] S. Mysore, G. Cheng, Y. Zhao, K. Saenko, and M. Wu, "Multi-critic actor learning: Teaching rl policies to act with style," in *International Conference on Learning Representations*, 2022.

[27] 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," in *Proceedings of Robotics: Science and Systems*, LosAngeles, CA, USA, June 2025.