

# PIE: Parkour with Implicit-Explicit Learning Framework for Legged Robots

Shixin Luo<sup>†1</sup>, Songbo Li<sup>†1</sup>, Ruiqi Yu<sup>1</sup>, Zhicheng Wang<sup>1</sup>, Jun Wu<sup>1,2</sup> and Qiuguo Zhu<sup>\*1,2</sup>

**Abstract**—Parkour presents a highly challenging task for legged robots, requiring them to traverse various terrains with agile and smooth locomotion. This necessitates comprehensive understanding of both the robot's own state and the surrounding terrain, despite the inherent unreliability of robot perception and actuation. Current state-of-the-art methods either rely on complex pre-trained high-level terrain reconstruction modules or limit the maximum potential of robot parkour to avoid failure due to inaccurate perception. In this paper, we propose a one-stage end-to-end learning-based parkour framework: Parkour with Implicit-Explicit learning framework for legged robots (PIE) that leverages dual-level implicit-explicit estimation. With this mechanism, even a low-cost quadruped robot equipped with an unreliable egocentric depth camera can achieve exceptional performance on challenging parkour terrains using a relatively simple training process and reward function. While the training process is conducted entirely in simulation, our real-world validation demonstrates successful zero-shot deployment of our framework, showcasing superior parkour performance on harsh terrains.

**Index Terms**—Legged robots, reinforcement learning, deep learning for visual perception.

## I. INTRODUCTION

**P**ARKOUR, an extremely challenging and exhilarating sport, encompasses a variety of movements such as running, jumping and climbing to traverse obstacles and terrain as quickly and smoothly as possible. Integrating quadruped robots into parkour entails overcoming numerous technical difficulties [1], [2], [3]. For robots, parkour requires a high degree of balance and stability to avoid falls or loss of control while moving rapidly and performing jumps. With obstacles and terrains constantly changing, robots need to accurately perceive their surroundings in real-time and make quick and

Manuscript received: April 23, 2024; Revised July 25, 2024; Accepted August 23, 2024.

This paper was recommended for publication by Editor Aleksandra Faust upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the National Key R&D Program of China (Grant No. 2022YFB4701502), the "Leading Goose" R&D Program of Zhejiang (Grant No. 2023C01177), the Key R&D Project on Agriculture and Social Development in Hangzhou City (Asian Games) (Grant No. 20230701A05), and the Key Research Project of Zhejiang Lab (Grant No. 2021NB0AL03).

<sup>†</sup>The authors are with Institute of Cyber-Systems and Control, Zhejiang University, 310027, China. {shixin\_luo, 3190105314, yrq, 3160105273}@zju.edu.cn, jwu@iipc.zju.edu.cn, qgzhuz@zju.edu.cn

<sup>2</sup>Qiangguo Zhu and Jun Wu are with State Key Laboratory of Industrial Control Technology, 310027, China.

<sup>†</sup>Shixin Luo and Songbo Li contributed equally to this work.

<sup>\*</sup>Qiangguo Zhu is the corresponding author.

The supplementary video is available at <https://youtu.be/XsjFNeND6js?si=9eLi8P3fTAXH1mc>.

Digital Object Identifier (DOI): see top of this page.



Fig. 1. Our parkour framework with implicit-explicit estimation allows low-cost robots to traverse a series of challenging parkour-like terrains. Notably, in the real world the robot is not limited to the terrain on which it was trained in simulation, demonstrating impressive generalization capabilities.

precise adjustments to avoid collisions or falls. However, successful implementation could open up new possibilities for robot technology in extreme environments and drive advancements in real-world applications and development.

Recent advancements in learning-based approaches have demonstrated significant strides in blind locomotion. Leveraging robust proprioceptive sensors, a quadruped robot can perform state estimation or system identification, achieving promising performance in walking, climbing stairs and mimicking animals [4], [5], [6], [7]. Although the robot can implicitly infer its surrounding terrains, it is still too difficult for a blind policy to foresee the upcoming dangers posed by the harsh terrains and make appropriate response immediately in the extreme tasks like parkour.

To tackle the complexities of varied terrain environments and accomplish more intricate tasks, exteroceptive sensors such as cameras and LiDARs have been employed, yielding advancements in various scenarios including locomotion and manipulation [8], [9], [10], [11], [12]. Recent studies indicate that even a low-cost quadruped robots equipped with a depth camera can achieve commendable results in parkour [3].

However, several challenges persist, limiting the maximal performance of quadruped robots in parkour: Firstly, exteroceptive sensors, although essential for enhancing performance, suffer from latency and noise, which poses challenges in achieving real-time and precise terrain estimation, crucial for maximizing robot performance in parkour where robots frequently encounter edges and must avoid falls. Secondly, existing learning-based parkour approaches usually adopt a

two-stage training paradigm, complicating the training process and potentially leading to information loss during mimicking learning phases, whether achieved through imitation learning or adaptation methods. Thirdly, integrating multiple behaviors seamlessly into a neural network via a relatively simple training process and reward function remains challenging. To address all these challenges, can we enhance the robot's understanding of its own state and surroundings through a simple one-stage training framework, pushing the limits of parkour further?

In this paper, we propose a one-stage end-to-end reinforcement learning-based framework: Parkour with Implicit-Explicit learning framework for legged robots (PIE), which elevates the parkour capabilities of quadruped robots to the next level through a simple yet effective training process. Motivated by recent works [3], [5], we introduce a hybrid dual-level implicit-explicit estimation approach. Firstly, on the level of understanding the robot's state and surroundings, PIE goes beyond the explicit understanding of the terrain through exteroceptive sensors. Instead, it integrates real-time and robust proprioception with exteroception to implicitly infer the robot's state and surroundings by estimating its successor state, which significantly enhances estimation accuracy. Furthermore, concerning whether the vector representing the robot's state and surroundings is an encoded latent vector or an explicit physical quantity, it explicitly estimates specific physical quantities together with latent vectors, which greatly enhances the quadruped robot's performance in executing parkour maneuvers on challenging terrains. We deployed PIE on a low-cost DEEP Robotics Lite3 quadruped robot and conducted various tests in parkour terrains and real-world scenarios, demonstrating remarkable performance and robustness.

In summary, our main contributions include:

- A novel one-stage learning-based parkour framework utilizing dual-level implicit-explicit estimation is proposed to enhance the quality of estimating the robot's state and surroundings.
- Experiments were conducted to show that our framework enables the robot to leap onto and jump off steps 3x its height, negotiate gaps 3x its length, and climb up and down stairs 1x its height. These results push the limits of parkour for the quadruped robot to a new level.
- Our real-world validation demonstrates effective zero-shot deployment of our framework without extensive fine-tune in the indoor environment and in the wild, showcasing great robustness of our sim-to-real transfer achieved through an elegant and potent pipeline.

## II. RELATED WORK

### A. Vision-Guided Locomotion

Vision-guided locomotion plays a pivotal role in enhancing the autonomy and adaptability of legged robots. Traditional approaches typically decouple this problem into two components: perception and controller. The perception component translates visual inputs into elevation maps or traversability maps to guide the robot's locomotion. Meanwhile, the controller component employs either model-based methods [13],

[14] or RL methods [8], [15]. Some approaches bypass explicit map construction, wherein both perception and controller utilize learning-based techniques to generate hierarchical RL methods [16], [17]. Most of these methods are primarily used for robot navigation and obstacle avoidance, rather than flexible adaptation to complex terrains, as the decoupling process ultimately results in information loss and system delays. Recently, vision-guided RL methods have generally employed end-to-end control systems, showing great promise in traversing complex terrains. For example, Agarwal *et al.* [10] designed a two-stage learning framework where a student policy, guided by a teacher policy with privileged information inputs, directly predicts joint angles based on depth camera inputs and proprioceptive feedback. Yang *et al.* [11] proposed a coupled training framework utilizing a transformer structure to integrate both proprioception and visual observations. Leveraging a self-attention mechanism, it fuses these inputs, enabling autonomous navigation in indoor and outdoor environments with varying obstacles. Yang *et al.* [12] utilized a 3D voxel representation with  $SE(3)$  equivariance as the extracted feature from visual inputs, achieving precise understanding of terrain.

### B. Robot Parkour

In the context of parkour, constructing precise environmental understanding through exteroceptive and proprioceptive sensors in highly dynamic situations to achieve agile, nimble and robust movement is paramount. Hoeller *et al.* [1] described a hierarchical pipeline for navigation in parkour terrains. However, when encountering unstructured terrain, occupancy voxels obtained from an encoder-decoder architecture often imply incorrectness and mistrust, leading to inappropriate responses from navigation and locomotion modules, along with high training costs and low scalability. Zhuang *et al.* [2] proposed a multi-stage method leveraging soft/hard constraints to accelerate the training process, enabling the robot to learn traversing various terrains directly from depth images. However, its privileged physics information is strongly associated with the geometric properties of obstacles in simulation. This makes it challenging for training the robot to cope with terrain that cannot be described solely by geometric properties. Cheng *et al.* [3] adopted a framework akin to [10]. Differently, it introduces waypoints into the teacher policy's privileged inputs to guide the student policy to autonomously learn heading, but it is the manual specification of waypoints based on terrain that imposes considerable limitations on this method. Furthermore, the two-stage training paradigm used by all aforementioned parkour works results in information loss and performance degradation of the deployed student policy. Meanwhile, these works primarily focus on explicit terrain estimation distilled from the teacher policy, and lack the implicit estimation of the surroundings, thereby hindering the robot's maximal performance capabilities.

## III. METHOD

The proposed PIE framework is a one-stage end-to-end learning-based framework that utilizes a single neural network

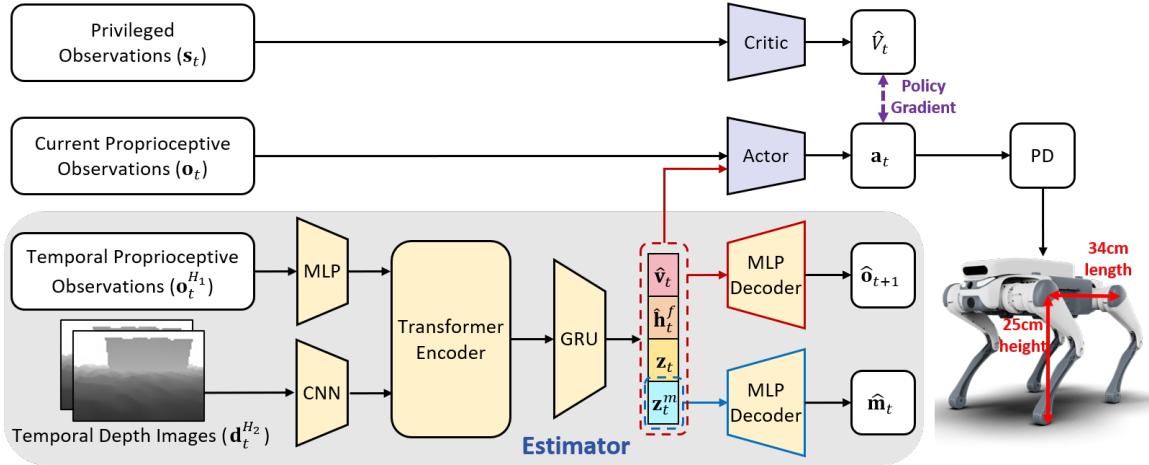


Fig. 2. Overview of the proposed PIE framework. The grey box represents the estimator, which utilizes implicit-explicit estimation to provide the policy network with the estimated vectors. The framework is concurrently optimized with PPO for the actor and critic network and regression for the estimator.

to directly derive desired joint angle commands from raw depth images and onboard proprioception. It circumvents the performance reductions associated with the two-stage training paradigm prevalent in prior learning-based parkour methodologies, aiming to enhance the estimation effectiveness of the robot's state and surroundings through dual-level implicit-explicit estimation. Here, we provide an overview of the framework, followed by detailed expositions.

### A. Overview

As shown in Fig. 2, the PIE framework comprises three sub-networks: actor, critic and estimator which will be elaborated upon in Section III-B. An asymmetric actor-critic architecture is adopted to simplify the two-stage training paradigm into a single stage [5]. The actor network receives inputs solely from proprioception and exteroception data obtainable during deployment, whereas the critic network can incorporate additional privileged information. The optimization process consists of two concurrent parts: the optimization of the actor-critic and the optimization of the estimator. In this work, the actor-critic is optimized using the proximal policy optimization (PPO) algorithm [18].

1) *Policy Network*: The policy network takes the proprioceptive observation  $\mathbf{o}_t$ , the estimated base velocity  $\hat{\mathbf{v}}_t$ , the estimated foot clearance  $\hat{\mathbf{h}}_t^f$ , the encoded height map estimation  $\mathbf{z}_t^m$  and the purely latent vector  $\mathbf{z}_t$  as input, producing action  $\mathbf{a}_t$  as output. The proprioceptive observation  $\mathbf{o}_t$  is a 45-dimensional vector directly measured from joint encoders and the IMU, which is defined as

$$\mathbf{o}_t = [\omega_t \quad \mathbf{g}_t \quad \mathbf{c}_t \quad \theta_t \quad \dot{\theta}_t \quad \mathbf{a}_{t-1}]^T, \quad (1)$$

encompassing the body angular velocity  $\omega_t$ , the gravity direction vector in the body frame  $\mathbf{g}_t$ , the velocity command  $\mathbf{c}_t$ , the joint angle  $\theta_t$ , the joint angular velocity  $\dot{\theta}_t$  and the previous action  $\mathbf{a}_{t-1}$ .  $\hat{\mathbf{v}}_t$ ,  $\hat{\mathbf{h}}_t^f$ ,  $\mathbf{z}_t^m$  and  $\mathbf{z}_t$  are the outputs of the estimator.

TABLE I  
REWARD FUNCTION ELEMENTS

Reward	Equation( $r_i$ )	Weight( $w_i$ )
Lin. velocity tracking	$\exp\{-4(\mathbf{v}_{xy}^{\text{cmd}} - \mathbf{v}_{xy})^2\}$	1.5
Ang. velocity tracking	$\exp\{-4(\omega_{yaw}^{\text{cmd}} - \omega_{yaw})^2\}$	0.5
Linear velocity ( $z$ )	$v_z^2$	-1.0
Angular velocity ( $xy$ )	$\omega_{xy}^2$	-0.05
Orientation	$ \mathbf{g} ^2$	-1.0
Joint accelerations	$\dot{\theta}^2$	$-2.5 \times 10^{-7}$
Joint power	$ \tau  \dot{\theta} $	$-2 \times 10^{-5}$
Collision	$-\frac{n_{\text{collision}}}{n_{\text{collision}}}$	-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

2) *Value Network*: To obtain a more accurate estimation of the state value  $\hat{\mathbf{v}}_t$ , the value network's input  $\mathbf{s}_t$  includes not only the proprioceptive observation  $\mathbf{o}_t$  but also the privileged observations base velocity  $\mathbf{v}_t$  and height map scan dots  $\mathbf{m}_t$ , which is defined as

$$\mathbf{s}_t = [\mathbf{o}_t \quad \mathbf{v}_t \quad \mathbf{m}_t]^T. \quad (2)$$

3) *Action Space*: The action  $\mathbf{a}_t$  is a 12-dimensional vector corresponding to the 12 joints of the quadruped robot. For the stability of the policy network output, the action  $\mathbf{a}_t$  is added as bias to the robot's standstill pose  $\theta_{\text{stand}}$ . Thus, the final target joint angle  $\theta_{\text{target}}$  is defined as

$$\theta_{\text{target}} = \theta_{\text{stand}} + \mathbf{a}_t. \quad (3)$$

4) *Reward Function*: To underscore the robust performance of our framework, we utilize a relatively simple reward function in a parkour scenario, closely aligned with prior research on blind walking [19], [20]. Details of the reward function are provided in Table I.  $\mathbf{g}$ ,  $\tau$  and  $n_{\text{collision}}$  are the gravity vector in the body frame, joint torque and the number of collision points other than the feet, respectively.

### B. Estimator

The vectors to be estimate,  $\hat{\mathbf{v}}_t$ ,  $\hat{\mathbf{h}}_t^f$ ,  $\mathbf{z}_t^m$  and  $\mathbf{z}_t$ , can be divided into implicit and explicit estimations on two levels.

On the level of understanding the robot's state and surroundings, a blind robot can implicitly infer its state and surrounding terrain solely through robust onboard sensors by estimating the successor proprioceptive state [5], [21]. However, it must interact with the environment and then adjust its inference based on immediate feedback. In the parkour scenario, the robot must anticipate the terrain ahead. For example, in high or long jumps, the robot needs to gather momentum in advance to execute liftoff successfully, which can only be achievable through exteroceptive sensors providing additional information. However, solely explicitly estimating terrain using laggy and noisy depth cameras is not sufficiently reliable, especially near edges where the risk of stepping off is high [2], [3]. Hence, we propose a multi-head auto-encoder mechanism to integrate implicit and explicit estimation of the robot's state and surroundings.

The encoder module has two inputs: temporal depth images  $\mathbf{d}_t^{H_2}$  and temporal proprioceptive observations  $\mathbf{o}_t^{H_1}$ . The depth observations in the near past are stacked in the channel dimension to form  $\mathbf{d}_t^{H_2}$ , while the proprioceptive observations in the near past are concatenated to form  $\mathbf{o}_t^{H_1}$ . In this work, we set  $H_1 = 10$  and  $H_2 = 2$ .  $\mathbf{d}_t^{H_2}$  is processed by a CNN encoder to extract depth feature maps, while  $\mathbf{o}_t^{H_1}$  is processed by an MLP encoder to extract proprioceptive features. To facilitate cross-modal reasoning between visual tokens from the 2D depth feature map and proprioceptive features, we employ a shared transformer encoder to further integrate the two modalities. Since the robot is equipped with only one egocentric depth camera, it cannot directly perceive terrain beneath or behind its body visually. Therefore, the outputs of the transformer, which are the depth and proprioceptive features after cross-modal attention processing, are concatenated and fed into a GRU to generate memories of the state and terrain. The outputs of the GRU are then served as the policy network's input vectors. One of these vectors is a encoded height map estimation vector  $\mathbf{z}_t^m$ , which is decoded into  $\hat{\mathbf{m}}_t$  to reconstruct the high-dimensional height map  $\mathbf{m}_t$ , serving as the explicit estimation of the surrounding terrain. Another vector is a purely latent vector  $\mathbf{z}_t$ . Instead of using a pure auto-encoder, following prior work [5], we adopt a VAE structure to extract features for  $\mathbf{z}_t$ . Namely, we employ the KL divergence as the latent loss. When decoded alongside other vectors with explicit meanings, it reconstructs  $\hat{\mathbf{o}}_{t+1}$  to represent the robot's successor state  $\mathbf{o}_{t+1}$ , encapsulating implicit information about the robot's state and surroundings. We use mean-squared-error (MSE) loss between the decoded reconstruction and the ground truth for both explicit and implicit estimations. This constitutes the first level of implicit-explicit estimation.

The second level concerns whether the vector is an encoded latent vector or an explicit physical quantity.  $\mathbf{z}_t^m$  and  $\mathbf{z}_t$  are encoded latent vectors followed by a decoder for reconstruction. Through this compression and dimension reduction, noise influence is reduced, resulting in greater robustness. For the explicit physical quantities,  $\hat{\mathbf{v}}_t$  and  $\hat{\mathbf{h}}_t^f$  are explicitly estimated to prioritize velocity tracking during training and provide relevant foot clearance information crucial for understanding terrain in parkour scenarios. The loss for this explicit physical quantities estimation part is also mean-squared-error (MSE)

TABLE II  
DOMAIN RANDOMIZATION RANGES

Parameter	Randomization range	Unit
Payload	$[-1, 2]$	kg
$K_p$ factor	$[0.9, 1.1]$	Nm/rad
$K_d$ factor	$[0.9, 1.1]$	Nms/rad
Motor strength factor	$[0.9, 1.1]$	Nm
Center of mass shift	$[-50, 50]$	mm
Friction coefficient	$[0.2, 1.2]$	-
Initial joint positions	$[0.5, 1.5]$	rad
System delay	$[0, 15]$	ms
Camera position ( $x$ )	$[-10, 10]$	mm
Camera position ( $y$ )	$[-10, 10]$	mm
Camera position ( $z$ )	$[-10, 10]$	mm
Camera pitch	$[-1, 1]$	deg
Camera horizontal FOV	$[86, 88]$	deg

loss. This constitutes the second level of implicit-explicit estimation.

The overall training loss for our estimator is defined as

$$\begin{aligned} \mathcal{L} = & D_{\text{KL}}(q(\mathbf{z}_t | \mathbf{o}_t^{H_1}, \mathbf{d}_t^{H_2}) \| p(\mathbf{z}_t)) + \text{MSE}(\hat{\mathbf{o}}_{t+1}, \mathbf{o}_{t+1}) \\ & + \text{MSE}(\hat{\mathbf{m}}_t, \mathbf{m}_t) + \text{MSE}(\hat{\mathbf{v}}_t, \mathbf{v}_t) + \text{MSE}(\hat{\mathbf{h}}_t^f, \mathbf{h}_t^f), \end{aligned} \quad (4)$$

where  $q(\mathbf{z}_t | \mathbf{o}_t^{H_1}, \mathbf{d}_t^{H_2})$  is the posterior distribution of the  $\mathbf{z}_t$ , given  $\mathbf{o}_t^{H_1}$  and  $\mathbf{d}_t^{H_2}$ .  $p(\mathbf{z}_t)$  is the prior distribution of  $\mathbf{z}_t$  parameterized by a standard normal distribution.

### C. Training Details

1) *Simulation Platform:* We utilize Isaac Gym with 4096 parallel environments to train the actor, critic and estimator in simulation. Leveraging NVIDIA Warp, we are able to train 10,000 iterations in under 20 hours on the NVIDIA RTX 4090, resulting in a readily deployable network.

2) *Training Curriculum:* Following the principles of prior works, we adopt a curriculum where the difficulty of the terrain progressively increased to enable the policy to adapt to increasingly challenging environments [19]. Our parkour terrains include gaps up to 1m wide, steps up to 0.75m height, hurdles up to 0.75m height and stairs with a height up to 0.25m. Lateral velocity commands are sampled from the range of  $[0.0, 1.5]$  m/s, and horizontal angular velocity are sampled from the range of  $[-1.2, 1.2]$  rad/s.

Additionally, to ensure that the deployed policy can traverse terrains beyond the fixed paradigms in simulation, besides manifesting variations in terrain difficulty, we introduce some randomization to various terrains in simulation. For instance, to enable the robot to adapt to the large gap which can be easily mistaken as a flat ground between two steps where it can jump down and then jump up to the next step, we randomize the depth and width of the gap.

3) *Domain Randomization:* To enhance the robustness of the network trained in simulation and facilitate smooth sim-to-real transfer, we randomize the mass of the robot body, the center of mass (CoM) of the robot, the mass payload, the initial joint positions, the ground friction coefficient, the motor strength, the PD gains, the system delay, as well as the camera position, orientation and field of view. The ranges for randomization of each parameter are specified in Table II.

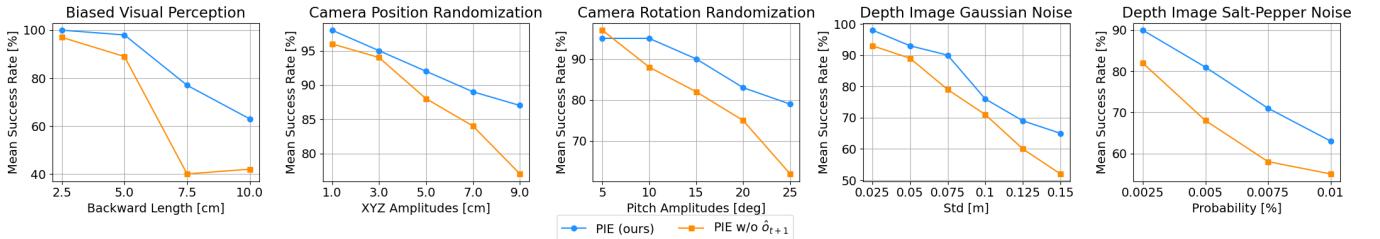


Fig. 3. Simulation experiments results for PIE and PIE without  $\hat{o}_{t+1}$  in the presence of various camera input errors. Five plots represent the five types of camera errors introduced, with the x-axis of each plot representing: the distance deviates of the depth image from the actual terrain perception in the world coordinate system along the negative x-axis; the maximum value of uniformly distributed noise added to the position of the camera relative to the base link of the robot in the x, y and z directions; the maximum value of uniformly distributed noise added to the pitch of the camera coordinate system relative to the robot coordinate system; the standard deviation of Gaussian noise added to the depth image; the amount of salt-pepper noise added to the depth image, with random pixels being set to either the minimum or maximum depth based on the probability. These experiments adopt the highest difficulty level of terrains encountered by the robot during training, with the y-axis representing the average success rate across all terrains, including gap, stairs and step. Similarly, to reflect the generality of the experimental results, the average results of one hundred robots are taken for each experiment.

#### IV. EXPERIMENTS

To evaluate the effectiveness of our framework, we conducted ablation studies in both simulation and real-world environments, and compared them with existing parkour frameworks. Here, we describe our experimental setup, present the experimental results and provide analysis.

##### A. Experimental Setup

We deployed PIE on a DEEP Robotics Lite3, which is a low-cost quadruped robot. As illustrated in Fig. 2, when standing, the thigh joint height is 25cm, and the distance between the two thigh joints measures 34cm. The Lite3 weighs 12.7kg, with a peak knee joint torque of 30.5Nm [22], [23]. Similarly, the Unitree A1, used by Zhuang *et al.* [2] and Cheng *et al.* [3], has a thigh joint height of 26cm and a distance of 40cm between the thigh joints [24]. It weighs 12kg, with a peak knee joint torque of 33.5Nm. Thus, it is reasonable to use their results as baselines in our comparative experiments. Raw depth images are received at a frequency of 10Hz using its onboard Intel RealSense D435i camera. Joint actions are outputted at a frequency of 50Hz. Image post-processing and network inference are performed using its onboard Rockchip RK3588 processor.

We designed five sets of ablations to compare against ours, aiming to verify the effectiveness of PIE’s dual-level implicit-explicit estimation:

- *PIE w/o reconstructing  $\hat{o}_{t+1}$* : This method can only estimate the robot’s state and surroundings explicitly without reconstructing  $\mathbf{o}_{t+1}$ .
- *PIE w/o reconstructing  $\hat{\mathbf{m}}_t$* : This method utilizes both proprioception and exteroception but only reconstructs  $\mathbf{o}_{t+1}$  as implicit estimation of surrounding, lacking explicit estimation of terrain.
- *PIE w/o estimating  $\hat{\mathbf{v}}_t$* : This method is trained without estimating  $\hat{\mathbf{v}}_t$ .
- *PIE w/o estimating  $\hat{\mathbf{h}}_t^f$* : This method is trained without estimating  $\hat{\mathbf{h}}_t^f$ .
- *PIE using predicted  $\mathbf{o}_{t+1}$* : This method allows the policy network to use the predicted  $\mathbf{o}_{t+1}$  as input, rather than the pure latent vector  $\mathbf{z}_t$  used for reconstruction.

TABLE III  
ABLATION STUDY FOR THE VECTORS TO BE ESTIMATED:  
PERFORMANCE IN SIMULATION

Method	Mean Terminated Difficulty Level		
	Gap	Stairs	Step
<b>PIE (ours)</b>	<b>9.9</b>	<b>9.86</b>	<b>9.81</b>
PIE w/o $\hat{o}_{t+1}$	9.51	9.45	9.62
PIE w/o $\hat{\mathbf{h}}_t^f$	7.41	7.36	3.09
PIE w/o $\hat{\mathbf{v}}_t$	8.7	8.22	8.48
PIE w/o $\hat{\mathbf{m}}_t$	9.75	4.25	1.67
PIE using predicted $\hat{o}_{t+1}$	9.23	7.28	3.29

##### B. Simulation Experiments

In simulation, we implemented an environment similar to the training setup for metric comparison, wherein we evaluated the performance of PIE against other ablations on different terrains. To precisely assess the policy’s traversal capability across terrains, including gap, stairs and step (hurdle was not specifically tested in simulation as the randomized step lengths covered the hurdle characteristics), we configured curriculum-like terrains with ten levels of increasing difficulty per column. We measured average level of terrain difficulty that robots can reach by letting robots start from the easiest level and gradually traversing the rest of their column until termination due to falls or collisions. For each terrain type, we created forty sets of environments, with one hundred robots tested simultaneously to ensure randomness in terrains and robots.

As shown in Table III, our PIE framework utilizing both implicit and explicit estimation demonstrates superior performance across all terrains. Although PIE w/o  $\hat{o}_{t+1}$  exhibits commendable performance in simulation, its reliance solely on explicit terrain estimation leads to a lack of comprehensive terrain understanding, making it susceptible to falls due to minor deviations in foothold positioning when encountering the most challenging terrains. The performance of PIE w/o  $\hat{\mathbf{h}}_t^f$  significantly declines compared to the former two methods because directly estimating  $\hat{\mathbf{h}}_t^f$  remains crucial for the robot’s direct comprehension of the terrain beneath its feet, lacking which can hinder the robot’s ability to execute extreme park-

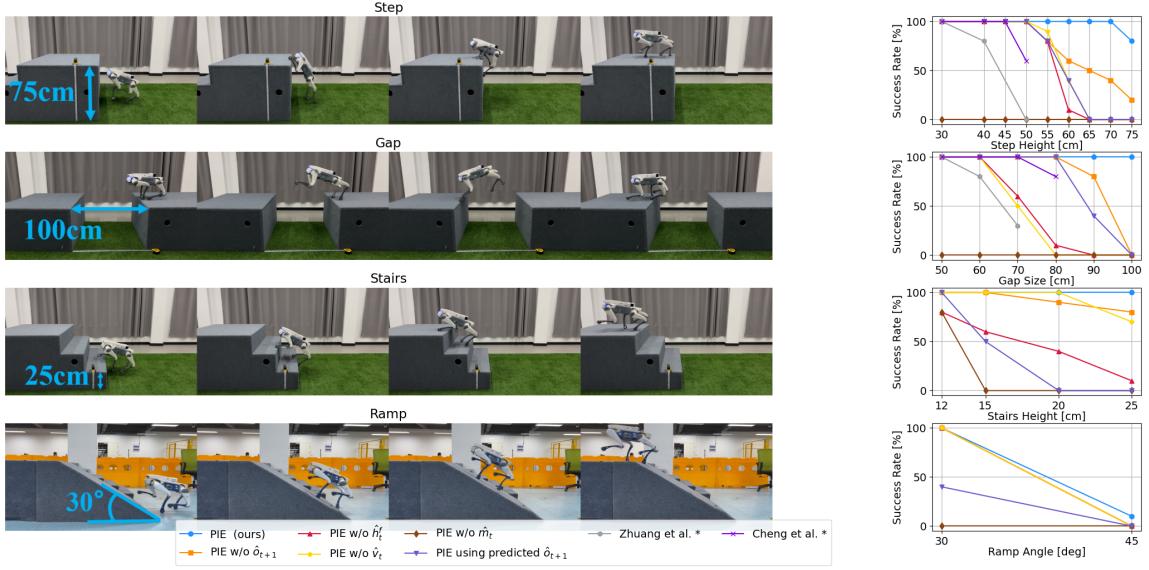


Fig. 4. Real-world indoor experiments results of each method. We measured the success rates for ten trials on each terrain of each difficulty level. Our PIE framework maintains consistent performance between real-world deployment and simulation, outperforming all other methods. Additionally, it demonstrates a certain level of generalization ability to traverse ramp terrains. Starred are previous related works [2], [3].

our maneuvers by stepping along the terrain edges. Without estimating the robot's velocity, the PIE w/o  $\hat{v}_t$  evidently introduces significant biases in velocity tracking, consequently diminishing the robot's maneuverability on high-difficulty terrains. The absence of regression between predicted  $\hat{m}_t$  and the ground truth in PIE w/o  $\hat{m}_t$  makes it difficult to extract useful terrain information directly from the input depth image, resulting in poor performance on highly challenging parkour terrains. PIE using predicted  $\hat{o}_{t+1}$  struggles on step and stairs terrain because unlike  $\hat{z}_t$ , which has a prior distribution parameterized by a standard normal distribution, the distribution of  $\hat{o}_{t+1}$  is more complex, making it more challenging for the policy network to extract useful information from it.

Additionally, we evaluated the performance of both PIE and PIE w/o  $\hat{o}_{t+1}$  across all terrains in the presence of various camera input errors by employing significantly higher noise levels to the camera, beyond those used in standard randomization. These errors normally cause discrepancies between visual inputs and the actual environment, which often occurs under conditions where the unreliable depth camera is subject to various interference. We employed these metrics to assess the effectiveness of learning implicit-explicit estimation in such scenarios as shown in Fig. 3. Overall, PIE outperforms PIE w/o  $\hat{o}_{t+1}$  in both metrics, particularly when considerable noise is added to the camera, resulting in significant performance disparities between the two. In contrast, within the normal domain randomization range, their performances are comparable. This suggests that not only can our learning framework better acquire robot state estimation and terrain understanding, but also, through the estimation of  $\hat{o}_{t+1}$ , the policy can make correct decisions when vision contradicts proprioception, placing more trust in proprioception to achieve better performance.

TABLE IV  
COMPARISON OF PARKOUR ABILITIES

Method	Robot	Step	Gap	Stairs
Hoeller <i>et al.</i> [1]	AnymalC	2×	1.5×	0.5×
Zhuang <i>et al.</i> [2]	Unitree-A1	1.6×	1.5×	-
Cheng <i>et al.</i> [3]	Unitree-A1	2×	2×	-
<b>PIE (ours)</b>	DEEP Robotics Lite3	<b>3×</b>	<b>3×</b>	<b>1×</b>

### C. Real-World Indoor Experiments

We compared the success rates of PIE with those of the ablations in the real world. Each method was tested for ten trials on each terrain of each difficulty level, including step, gap and stairs. Additionally, although we did not train specifically on ramp terrains, we still conducted tests on ramp terrains to evaluate the generalization performance.

As shown in Fig. 4 and the supplementary video, owing to the implicit-explicit estimation, our framework shows outstanding performance across all skills, surpassing all other ablations and previous related works. We find that our PIE framework enables the robot to climb obstacles as high as 0.75m (3x robot height), leap over gaps as large as 1m (3x robot length) and climb stairs as high as 0.25m (1x robot height). This represents a significant performance improvement of at least 50% compared to the state-of-the-art robot parkour frameworks as shown in Table IV, which indicates the relative size of traversable obstacles with respect to quadruped's height and length respectively. Notably, PIE exhibits consistent success rates compared to in simulation, demonstrating remarkable sim-to-real transferability. It is worth mentioning that, despite the lack of specific training for ramp terrains in simulation, our framework still demonstrates better generalization performance.

Among the ablations, PIE w/o  $\hat{o}_{t+1}$  performs relatively

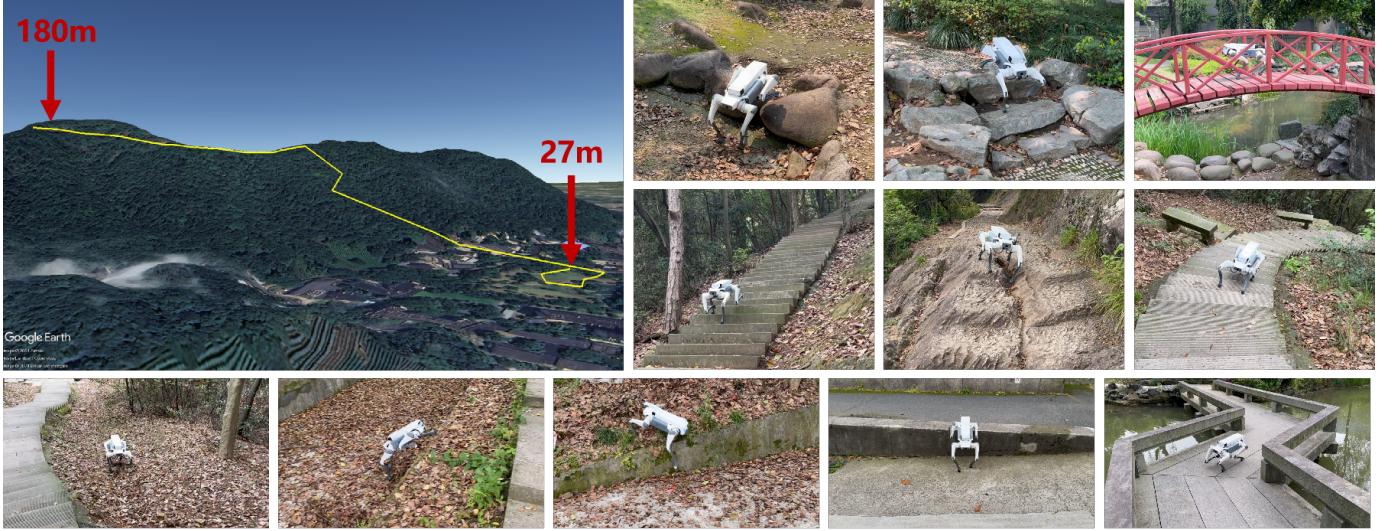


Fig. 5. A 2km round-trip hike from the ZJU Yuquan campus to the top of Laohe Mountain and back, with an elevation gain of 153 meters from 27m to 180m. Our PIE framework successfully navigated the challenging terrains along this route, demonstrating notable generalization capabilities.

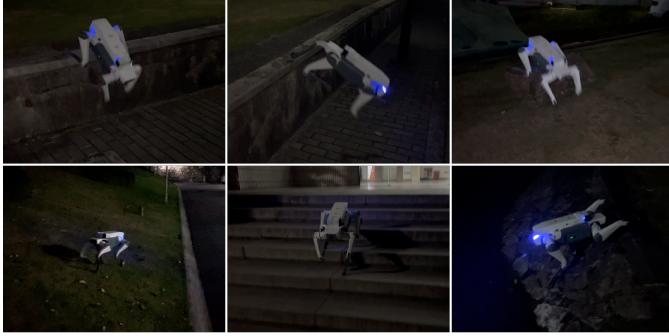


Fig. 6. Tests in dark outdoor environment. Despite the near absence of visible light, the robot was able to accurately perform agile maneuvers.

better, but due to the larger delay and noise in perception and actuation in the real world, its performance noticeably decreases compared to in simulation. PIE w/o  $\hat{h}_t^f$  struggles to learn to handle terrain edges correctly due to the lack of intuitive estimation of foot clearance, resulting in lower success rates. As for PIE w/o  $\hat{v}_t$ , when the terrain difficulty increases, the estimation of base velocity deteriorates, leading to a noticeable decline in success rates when following velocity commands. PIE w/o  $\hat{m}_t$  faces challenges extracting useful terrain information from more complicated real-world depth image compared to in simulation, making external perception interference rather than assistance, thus resulting in near-zero success rates across all terrains.

#### D. Real-World Outdoor Experiments

Due to the significant disturbances the depth camera faces in outdoor settings, the sim-to-real gap becomes more pronounced. To fully assess the robustness and generalization abilities of the PIE framework in such environments, as shown in Fig. 5, we conducted a long-distance hike from the ZJU Yuquan campus to the top of Laohe Mountain and back. The round trip covered 2km, with an elevation gain

of 153m. Along this trail, the robot was required to traverse long-range continuous curved stairs of varying heights and widths, irregularly shaped steps and hurdles, ramps with steep inclinations, as well as deformable, slippery ground and rocky surfaces. The robot completed the entire hiking trail in just 40 minutes with no stops, except for moments when the operator and photographer struggled to keep up due to the robot's rapid ascent on inconsistently high and wide stairs. Additionally, as shown in Fig. 6, we conducted tests in dim outdoor conditions at night, where the robot continuously jumped over high steps and irregular rocks, and climbed up and down slopes and stairs. Please refer to the supplementary video for the demonstration of the hike and night tests.

## V. DISCUSSION

Despite not employing imitation learning or designing intricate reward functions to constrain robot behaviors, our straightforward training process enables the robot to achieve a natural and agile gait, seamlessly transitioning across various complex terrains, similar to the natural movements observed in real cats and dogs. We hypothesize that this is attributed to its ability to predict the successor state, which not only enhances its capability to manage delays [25], but also establishes an internal model that fosters a better understanding of both itself and its environment [21].

Furthermore, in emergent scenarios where slight deviations in external perception occur—such as being momentarily tripped up while running up 25cm stairs at high speed, or misstepping during the takeoff phase of a 1m gap jump, which would typically be catastrophic even for a human traceur—we were surprised to observe that our PIE framework was capable of executing timely and accurate responses, which is shown in Fig. 7 and the supplementary video. This is particularly noteworthy given that reinforcement learning is generally not adept at handling precise maneuvers, making such deviations difficult to avoid entirely.

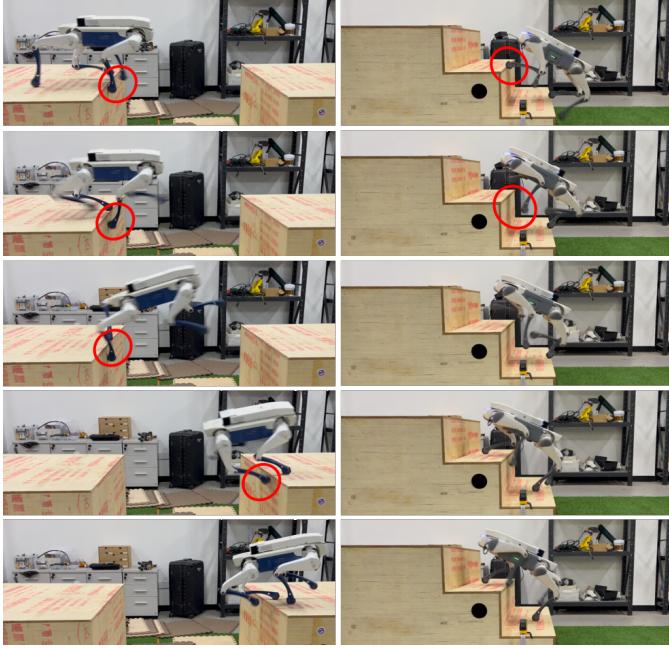


Fig. 7. The robot can quickly regain stability even when estimations are inaccurate during intense maneuvers. We observed that when leaping over a gap, despite slight distance estimation errors causing the robot's front and rear legs to not fully support on the platform before takeoff, it still managed to execute the jumping action and land smoothly on the other side. Similarly, when encountering a sudden step void while ascending stairs, the robot promptly stabilized itself to continue stepping upwards.

However, certain limitations still exist. Firstly, it lacks a 3D understanding of the terrain, thus unable to crouch under obstacles. Secondly, our external perception relies solely on depth images, lacking the richer semantic information provided by RGB images. Additionally, our training is confined to static environments, without extension to dynamic scenes, potentially leading to confusion in visual estimation.

## VI. CONCLUSION

In this work, we propose a novel one-stage end-to-end learning-based parkour framework PIE, utilizing dual-level explicit-implicit estimation to refine the understanding of both the robot's state and its environment. Compared to existing learning-based parkour frameworks, PIE significantly enhances the parkour capabilities of robots with a unified policy. Meanwhile, PIE demonstrates commendable generalization in outdoor environments, maintaining effective performance across diverse conditions.

In the future, we aim to design a new unified learning-based sensorimotor integration framework, which extracts 3D terrain information from depth images and obtains abundant semantic information from RGB images, to achieve better adaptability and mobility in various environments.

## REFERENCES

- [1] 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.
- [2] Z. Zhuang, Z. Fu, J. Wang, C. Atkeson, S. Schwertfeger, C. Finn, and H. Zhao, "Robot parkour learning," *arXiv preprint arXiv:2309.05665*, 2023.
- [3] X. Cheng, K. Shi, A. Agarwal, and D. Pathak, "Extreme parkour with legged robots," *arXiv preprint arXiv:2309.14341*, 2023.
- [4] J. Hwangbo, J. Lee, A. Dosovitskiy, D. Bellicoso, V. Tsounis, V. Koltun, and M. Hutter, "Learning agile and dynamic motor skills for legged robots," *Science Robotics*, vol. 4, no. 26, p. eaau5872, 2019.
- [5] 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.
- [6] G. Ji, J. Mun, H. Kim, and J. Hwangbo, "Concurrent training of a control policy and a state estimator for dynamic and robust legged locomotion," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4630–4637, 2022.
- [7] 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.
- [8] 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.
- [9] X. Cheng, A. Kumar, and D. Pathak, "Legs as manipulator: Pushing quadrupedal agility beyond locomotion," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 5106–5112.
- [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] R. Yang, M. Zhang, N. Hansen, H. Xu, and X. Wang, "Learning vision-guided quadrupedal locomotion end-to-end with cross-modal transformers," *arXiv preprint arXiv:2107.03996*, 2021.
- [12] R. Yang, G. Yang, and X. Wang, "Neural volumetric memory for visual locomotion control," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 1430–1440.
- [13] P. Fankhauser, M. Bloesch, C. Gehring, M. Hutter, and R. Siegwart, "Robot-centric elevation mapping with uncertainty estimates," in *Mobile Service Robotics*. World Scientific, 2014, pp. 433–440.
- [14] P. Fankhauser, M. Bloesch, and M. Hutter, "Probabilistic terrain mapping for mobile robots with uncertain localization," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3019–3026, 2018.
- [15] S. Gangapurwala, M. Geisert, R. Orsolino, M. Fallon, and I. Havoutis, "Real-time trajectory adaptation for quadrupedal locomotion using deep reinforcement learning," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 5973–5979.
- [16] D. Jain, A. Iscen, and K. Caluwaerts, "From pixels to legs: Hierarchical learning of quadruped locomotion," *arXiv preprint arXiv:2011.11722*, 2020.
- [17] S. Katreer, N. Yokoyama, D. Batra, S. Ha, and J. Truong, "Vinl: Visual navigation and locomotion over obstacles," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 2018–2024.
- [18] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.
- [19] N. Rudin, D. Hoeller, P. Reist, and M. Hutter, "Learning to walk in minutes using massively parallel deep reinforcement learning," in *Conference on Robot Learning*. PMLR, 2022, pp. 91–100.
- [20] 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.
- [21] J. Long, Z. Wang, Q. Li, J. Gao, L. Cao, and J. Pang, "Hybrid internal model: A simple and efficient learner for agile legged locomotion," *arXiv preprint arXiv:2312.11460*, 2023.
- [22] "Deeprobotics lite3," <https://www.deeprobotics.cn/en/index/product1.html>, accessed on 2024-07-22.
- [23] "Deeprobotics j60 joint," <https://www.deeprobotics.cn/en/index/j60.html>, accessed on 2024-07-22.
- [24] "Unitree a1," <https://www.unitree.com/a1>, accessed on 2024-07-22.
- [25] W. Wang, D. Han, X. Luo, and D. Li, "Addressing signal delay in deep reinforcement learning," in *The Twelfth International Conference on Learning Representations*, 2023.