2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

PA-LOCO: Learning Perturbation-Adaptive Locomotion for Quadruped Robots

Zhiyuan Xiao, Xinyu Zhang, Xiang Zhou, and Qingrui Zhang

***Abstract*— Locomotion control is still a challenging task for quadruped robots traversing diverse terrains amidst unforeseen disturbances. Recently, privileged learning has been employed to learn reliable and robust quadrupedal locomotion over various terrains based on a teacher-student architecture. However, its one-encoder structure is not adequate in addressing external force perturbations. The student policy would experience inevitable performance degradation due to the feature embedding discrepancy between the feature encoder of the teacher policy and the one of the student policy. Hence, this paper presents a privileged learning framework with multiple feature encoders and a residual policy network for robust and reliable quadruped locomotion subject to various external perturbations. The multi-encoder structure can decouple latent features from different privileged information, ultimately leading to enhanced performance of the learned policy in terms of robustness, stability, and reliability. The efficiency of the proposed feature encoding module is analyzed in depth using extensive simulation data. The introduction of the residual policy network helps mitigate the performance degradation experienced by the student policy that attempts to clone the behaviors of a teacher policy. The proposed framework is evaluated on a Unitree GO1 robot, showcasing its performance enhancement over the state-of-the-art privileged learning algorithm through extensive experiments conducted on diverse terrains. Ablation studies are conducted to illustrate the efficiency of the residual policy network.**

I. Introduction

Model-free reinforcement learning method has demonstrated remarkable success in the advancement of locomotion controllers for legged robots [1]–[3]. Previous research aimed to enhance the blind locomotion of legged robots on various complex terrains, such as steps, slopes, grass, mud, snow, and sand, maximizing their potential for outdoor operation. Due to the complexity and variability of the missions and working environments, quadruped robots are vulnerable to various unexpected perturbations or disturbances, such as collisions with dynamic obstacles or external sudden forces. Hence, it is critical to efficiently deal with these disturbances to endow quadruped robots with safe and reliable locomotion capabilities.

However, safe and reliable locomotion is a challenging task in the presence of unexpected perturbations, especially in case without any force sensors. If left unaddressed, unforeseen perturbations, such as impact forces or sudden pushes, would propel a quadruped robot away from its stable locomotion, thus pushing the robot away from its intended trajectory. In severe scenarios, these perturbations can significantly deteriorate the locomotion stability of a quadruped robot, ultimately causing it to fall over. Therefore, it is beneficial, though challenging, for quadrupedal robots to actively compensate for such perturbations by using measurements from off-the-shelf onboard sensors, *e.g.*, an inertial measurement unit (IMU) and joint encoders *etc*.

Fig. 1: A quadruped robot is subject to a kick.

One straightforward idea is to learn a robust locomotion control policy using reinforcement learning by the so-called domain randomization technique [4]. It involves training a policy on various environments with randomized properties, such as perturbations of different magnitudes or sudden pushes from different directions. Through this process, the robot learns a robust, yet passive and conservative policy applicable across diverse conditions. A more effective solution, however, is to allow robots to react to external disturbances in an adaptive fashion using certain estimates based on onboard sensors.

As an alternative, privileged learning provides a viable solution to learning a locomotion control policy that can actively handle external disturbances via certain estimates. In privileged learning, a teacher-student structure is employed. The teacher policy is trained with additional or privileged information unavailable during testing or deployment. Such privileged information is embedded in a certain latent feature space. The latent feature space represents a lower-dimensional representation of the data that captures the underlying structure or patterns. A student policy is thereafter trained via supervised learning to imitate the behaviors of the teacher policy using available observations in real implementations [3], [5]–[7]. Privileged learning has been used to learn reliable and stable locomotion control policies for quadruped robots to

This work is supported by State Key Laboratory of Robotics and Systems (HIT) under Grant SKLRS-2024-KF-05, in part by Shenzhen Science and Technology Program JCYJ20220530145209021, and in part by Guang Dong Basic and Applied Basic Research Foundation under Grant 2024A1515012408.

All authors are with School of Aeronautics and Astronautics, Shenzhen Campus of Sun Yat-sen University, Shenzhen 518107, P.R. China. Correspondence to: Qingrui Zhang (zhangqr9@mail.sysu.edu.cn)

traverse challenging terrains with privileged information, such as terrain profiles, ground friction, and various robot states, including trunk mass and velocity. However, the potential of the privileged learning method remains unexplored for the learning of a locomotion controller adaptable to external perturbations.

Furthermore, the student policy commonly suffers from performance degradation in comparison with the teacher policy in the privileged learning framework. The behavior cloning process for student policy learning would naturally result in certain discrepancies between latent features inferred from available measurements and those from privileged information. Hence, the student policy is expected to be refined again. Another observation is that the single encoder architecture in the existing privileged learning is not adequate enough for perturbation compensation. With the one-encoder architecture, privileged perturbation information intertwines with other privileged signals. It is, therefore, impossible to distinguish the latent feature changes due to external perturbations from those by other variables (*e.g.*, velocities, or heading angle).

To tackle the aforementioned problems, this paper presents a teacher-student framework with multiple encoders as depicted in Fig. 2. The proposed framework aims to 1) achieve blind locomotion over diverse terrains; and 2) actively compensate for external perturbations using existing onboard measurements. Through this framework, the learned policy is more robust against sudden force disturbances in comparison with the state-of-the-art privilege learning algorithm. It also takes less time for the robot to recover its locomotion after the force impact. Experiments illustrate that the proposed framework can generate steady, adaptive, and robust locomotion in diverse perturbations and varied terrains. The overall contributions are three-fold:

1) A residual policy network is introduced to alleviate the student's performance degradation issue. The ablation study has shown that the residual policy network can improve the locomotion robustness and reduce the recovery time in the presence of disturbances.
2) The privileged learning is improved by using a multi-encoder structure. With this modification, latent features from different privileged information are decoupled from each other, which reduces the potential mutual influence among different observations. Experiments have demonstrated that the multi-encoder structure is beneficial to the improvement of policy performance, *e.g.,* robustness, stability, and reliability, *etc*.
3) The effectiveness of the latent feature embedding is analyzed sufficiently using simulated data. Extensive numerical simulations are performed to illustrate the effectiveness of the force encoder in distinguishing external forces of varying magnitudes and directions.

II. Related Works

A. Reinforcement learning-based control

In recent years, RL has been successfully implemented to deal with quadruped locomotion [8]. The concurrently trained

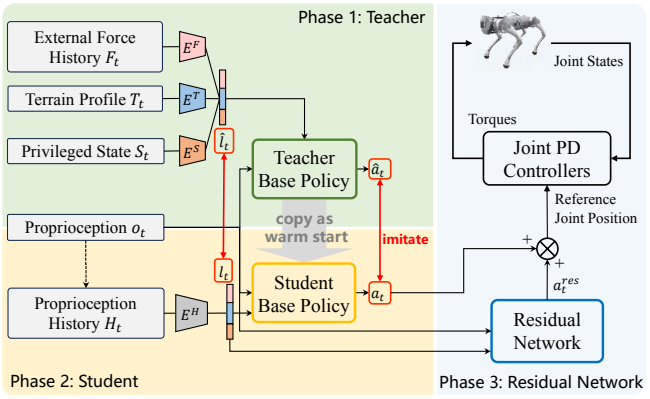

Fig. 2: The proposed PA-LOCO integrates a teacher-student framework with a residual network and multiple feature encoders, and its training process consists of three phases. In the first phase, the teacher policy is trained using proprioceptive observations o_t along with privileged information F_t, T_t, S_t. In the second phase, the student policy is trained with proprioceptive observations and is learned to replicate the teacher's actions and latent features. In the third phase, a residual policy network is trained to further improve the student policy's performance against perturbations.

control policy and state estimator are capable of traversing slippery grounds and bumpy roads [9]. RL-trained networks also demonstrated steady and reliable locomotion capabilities on challenging and deformable terrains [3], [10]. In addition, the learning-based method has been applied to other agile locomotion or manipulation skills, including fall recovery [2], [11], jumping [12], [13], parkouring [14], rotating balls with limbs [15], opening doors [16], and playing soccer [17]. In this paper, we focus on robust and adaptive quadruped locomotion control at various terrains in the presence of perturbations.

B. Policy adaptation

The domain randomization method makes it possible to directly deploy the trained robust policy on a real robot without any modification. Peng *et al.* achieved policy transfer for robust robotic arm operations via a domain randomization technique that randomly changes the parameters of the dynamic model in training [4]. Furthermore, Tan *et al.* conducted an analytical evaluation of domain randomization efficacy within quadruped locomotion contexts [1]. However, it requires a trade-off between robustness and performance. To achieve adaptation to a new environment, Peng *et al.* map the encoded dynamic parameters to a Gaussian distribution over a latent space [18]. However, the latent encoder needs to be re-trained in an offline fashion, when a robot is in a new environment with different settings. Recent advancements introduce real-time policy adaptation mechanisms, notably privileged learning and rapid motor adaptation (RMA).

C. Teacher-student learning framework

Privileged learning is a teacher-student learning framework, which is proposed by Lee *et al.* for the locomotion control of quadruped robots [3]. The privileged learning encodes

privileged terrain information, into a latent representation to solve the partial observability problem in blind quadruped locomotion [3]. It involves two phases of training. In the first phase, the teacher policy is trained using privileged information that is not available for deployment. In the second phase, the student policy is trained to replicate the teacher's behaviors using onboard sensor measurements that are easily accessible in real life. Kumar *et al.* propose a comparable RMA framework enabling the real-time online policy adaptation to novel situations within fractions of a second [5]. Additionally, the teacher-student learning framework provides a viable solution to infer privileged states using onboard sensor measurements [19]. Luo *et al.* utilize the parameterized motor failure as privileged information to implicitly identify faults [20]. However, the existing algorithms are not satisfactory in training a perturbation-adaptive locomotion control policy. In this paper, the aforementioned issues will be addressed.

III. Reinforcement Learning

Reinforcement Learning (RL) serves as a data-driven method that formulates the locomotion control problem within the framework of a Markov Decision Process (MDP). A parameterized control policy is learned through substantial trial and error using data from either simulations or the real world. Blind quadrupedal locomotion control, however, is more accurately modeled as a Partially Observable Markov Decision Process (POMDP). To address the issue of partial observability, privileged learning presents a promising framework by embedding unavailable privileged information into a latent feature space, which can be implicitly inferred using proprioceptive sensor measurements during real-world deployment. This approach enables the development of a reliable policy for robust and steady quadruped locomotion that is resilient to external disturbances.

Observations: The observation space of the teacher policy comprises proprioceptive sensor measurements, robot states, and external disturbances, as well as terrain profiles. The proprioceptive sensor measurements $o_t \in \mathbb{R}^{45}$ include trunk angular velocity ω_b obtained from the IMU, gravity unit vector in the body frame $\hat{g}$, joint positions $\{q_0, q_1, ..., q_{11}\}$, joint velocities $\{\dot{q}_0, \dot{q}_1, ..., \dot{q}_{11}\}$ output by the joint encoder, high-level commands $\{v_x^*, v_y^*, \omega_z^*\}$, and the actions at the last timestep. The robot state information $S_t \in \mathbb{R}^{28}$ includes trunk linear velocity, trunk mass, center of mass (COM), ground friction coefficient, foot contact forces with the ground, and contact states on the thigh and calf. Time-series information $F_t \in \mathbb{R}^{30}$ captures the external force disturbances over the last 10 timesteps. The terrain profile $T_t \in \mathbb{R}^{187}$ consists of 187 height values sampled below the robot's trunk. The observation space of the student policy only contains proprioceptive sensor measurements H_t.

Actions: The action space contains 12-dimensional reference joint angles $a_{RL,t}$ for a quadruped robot. The position command signal for each joint, q_t^*, is the sum of the default constant joint position $q_{default}$ and the RL output $a_{RL,t}$, so $q_t^* = q_{default} + a_{RL,t}$. The command signal q_t^* is sent to low-level PD controllers with proportional and derivative gains are $K_p = 20$ and $K_d = 0.5$, respectively.

TABLE I: Parameter setting for training

	Parameter	Range
Randomized dynamics	Trunk mass	[-1,1] kg
	COM displacement	[-0.03, 0.03] m
	Ground friction coefficient	[0.25, 1.5]
Trunk impulse	External force magnitude F_x, F_y	[-60, 60] N
	External force magnitude F_z	[-10, 10] N
	External force noise	[-2,2] N
	External wrench magnitude ω	2.5 rad/s
	External wrench interval	15 s
Sensor noises	Trunk angular velocity noise	[-0.05, 0.05] rad/s
	Gravity vector noise	[-0.05, 0.05]
	Joint positions noise	[-0.01, 0.01] rad
	Joint velocities noise	[-0.075, 0.075] rad/s

RL policy and reward shaping: Proximal Probability Optimization (PPO) is selected to learn the feedback body controller in our design [21]. In addition, we adopt the same reward terms from our previous work to encourage natural locomotion patterns [22].

Domain randomization: Domain randomization techniques are used during training to alleviate the sim-to-real gap. Dynamic parameters, including body mass, COM displacement, and ground friction, are randomized in each episode to simulate various environments. Random forces and torques are also applied during locomotion, and noise is added to sensor feedback to enhance the controller's robustness. The parameters for domain randomization, shown in TABLE I, follow a uniform distribution.

Physical simulator and training setup: Training is conducted in the Isaac Gym environment [23]. The quadruped robot is trained to follow high-level commands under random disturbances across various terrains with 4096 agents running in parallel. The simulation runs at 200 Hz, while the policy runs at 50 Hz. Each episode lasts up to 20 seconds (or 1000 time steps) and restarts if the time limit is reached or the robot's trunk collides with the ground. Omnidirectional velocity commands are uniformly sampled every 10 seconds.

IV. Teacher-Student Architecture

Teacher Policy: The teacher policy consists of two types of MLP networks, namely the base policy network and the encoder networks. The encoder networks include external force encoder E^F, terrain encoder E^T, and state encoder E^S, each of which has direct access to the corresponding privileged information. The structure of each network is listed in TABLE II.

Student Policy: The student policy comprises a base policy network π^S and a proprioception history encoder E^H. The

TABLE II: Network configurations

Module (MLP)	Input	Hidden Layers	Output
π^T	$o_t, \hat{l}_t$	[512, 256, 128]	$\hat{a}_t$
V	o_t, F_t, S_t, T_t	[512, 256, 128]	V_t
E^F	F_t	[64, 32]	$\hat{l}_t^F$
E^T	T_t	[256, 128]	$\hat{l}_t^T$
E^S	S_t	[64, 32]	$\hat{l}_t^S$
π^S	o_t, l_t	[512, 256, 128]	a_t
E^H	H_t	[1024, 512, 256]	l_t
R	o_t, l_t	[256, 128, 64]	a_t^{res}

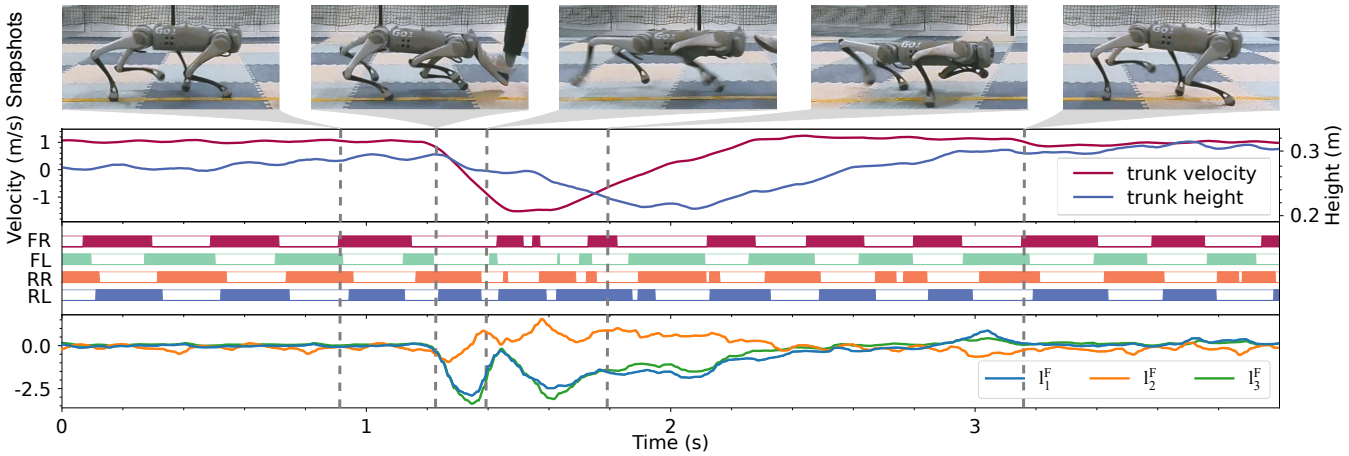

Fig. 3: The locomotion behavior under front-directed force impulses. The first image shows snapshots of the robot, the second displays trunk velocity and height responses, the third illustrates feet contact patterns (F/R for Front/Rear, R/L for Right/Left), and the bottom image presents force latent variable plots from the force encoder.

encoder receives the feedback from the proprioceptive sensor collected in the last second, *i.e.* the last 50 time steps. In the second phase, the student policy learns to adapt to different scenarios by imitating the teacher's actions $\hat{a}_t$ and latent features $\hat{l}_t = (\hat{l}_t^T, \hat{l}_t^S, \hat{l}_t^F)$. The student's base policy first copies the parameters of the teacher's base policy as a warm start. In addition, the training data is generated by rolling out the student's trajectories in simulation. The base policy and encoder are trained via supervised learning as in (1)

$$L = (\hat{a}_t - a_t)^2 + \left(\hat{l}_t - l_t\right)^2, \tag{1}$$

Residual Network: The residual network is an MLP that inputs proprioception data o_t and latent feature l_t to generate signals that enhance the student's locomotion under perturbations. As shown in Table II, this network is smaller than the student policy. Throughout the training phase, the parameters of each encoder and the student's base policy are kept frozen. Analogous to the training process for the teacher policy, an asymmetric actor-critic architecture is employed, with the critic network initiated by the teacher's counterpart. The resultant RL signal is a weighted sum of the outputs of the student policy and the residual network with weights of 0.25 and 0.1, respectively.

Curriculum Learning: A curriculum learning strategy, similar to [2], improves locomotion performance by gradually increasing perturbations during training. As the mean tracking reward reaches a threshold, the impulse magnitude rises. Additionally, a terrain curriculum enhances the robot's ability to traverse complex terrain. Training begins on flat terrain, and once the robot effectively tracks varying speed commands, it advances to more challenging terrains.

V. Experimental Results

Real-world experiments are conducted to validate the efficiency of the proposed PA-LOCO. The locomotion performance is evaluated under various external perturbations and terrain configurations. Additionally, simulation experiments are performed to analyze the efficiency of the latent representations of external forces with varying magnitudes and directions.

A. Adaptive locomotion under external perturbations

To evaluate locomotion adaptation and resilience under external perturbation, we kick the robot from the front while it tracks a constant velocity of 1 m/s, as shown in Fig. 3. Before the kick, the robot maintains a steady 1 m/s while tracking the constant velocity command. Its trunk height stabilizes at 0.29 m, as shown in Fig. 3, and the force encoder outputs remain near zero with slight variation. Upon being kicked, the robot sharply decelerates, dropping its forward velocity by -1.5 m/s in under half a second, as shown in Fig. 3. The external force encoder effectively captures this change. To withstand the sudden perturbation, the robot lowers its COM, increases foot contact frequency, and adopts non-structured gait patterns for enhanced stability. After the kick, the robot gradually resumes its motion, restoring its speed to 1 m/s and the trunk height to 0.3 m with slight oscillations. The external force encoder eventually returns to 0 as the impact force disappears.

B. Effects of force adaption and residual network

We compare our framework's performance with several benchmarks using a real-world quadruped robot (TABLE III). The analysis is performed as the robot traverses on a plane

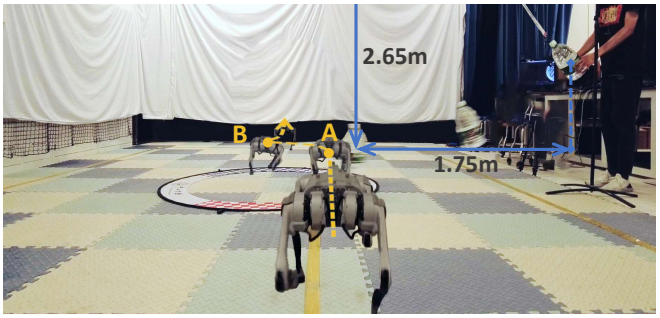

Fig. 4: Indoor experiment setup.

TABLE III: Metric results for different algorithms

Weights	Metrics	Algorithms			
		Robust	SEFI	MEFI	PA-LOCO
2.2 kg	SR (%)	90 (9/10)	0	**100** (10/10)	**100** (10/10)
	LO (m)	0.33	-	0.17	**0.16**
	RT (s)	1.06	-	0.50	**0.45**
	HO (m)	0.00	-	-0.01	**−0.02**
4.6 kg	SR (%)	43 (3/7)	0	80 (8/10)	**90** (9/10)
	LO (m)	1.46	-	0.64	**0.59**
	RT (s)	1.98	-	0.89	**0.75**
	HO (m)	0.00	-	**−0.06**	-0.05

at 1 m/s and encounters a sudden lateral impact. Alongside the proposed PA-LOCO, three benchmarks are trained for comparison:

- **Robust**: The policy is trained without privileged force information FI and a residual network R, but with domain randomization.
- **SEFI**: The policy uses a single-encoder structure SE and is trained with privileged force information.
- **MEFI**: An ablation study retaining the multi-encoder ME with privileged force information but removing the residual network R.

A pendulum system with a 2.65-meter arm and a suspended weight is used to generate lateral impact forces, as shown in Fig. 4. In each trial, the weight is raised so its center of mass (COM) is 1.75 meters horizontally from the pivot, maintaining a constant angle. At the lowest point, the COM is approximately 0.3 meters above the ground, matching the robot's trunk height, and collides with the robot's torso. Using 2.2 kg and 4.6 kg weights, the system induces lateral impacts, causing instantaneous velocity shifts of about 1.3 m/s and 2 m/s, respectively.

The following metrics are introduced to evaluate the performance of each algorithm: (1) Success Rate (SR): The rate of surviving lateral impacts without falling, which evaluates the robot's robustness against external disturbances; (2) Lateral COM Offset (LO): The average lateral displacement of the robot's center of mass due to impact, further assessing robustness; (3) Recovery Time (RT): The average time taken by the robot to recover after an impact, measuring its recovery capability; (4) Trunk Height Offset (HO): The average change in trunk height post-impact, assessing the robot's adaptation capability.

The results for the average trunk height offset are listed in TABLE III. The robust policy shows limited capability in adjusting trunk height in response to external impacts. In contrast, our PA-LOCO approach actively lowers trunk height upon impact. In scenarios with more severe impact forces from heavier weights, the force adaptation module enables the robot to lower its COM even further, enhancing stability.

The performance metrics listed in TABLE III confirm the efficiency of the force adaptation module and the add-on residual network. SEFI's performance cannot be assessed due to its highly unstable locomotion. The robust policy shows limited adaptability to unexpected perturbations, resulting in greater lateral movements triggered by impulses, as indicated by the LO metric. In contrast, both MEFI and PA-LOCO demonstrate lower LO and RT values, suggesting that policies

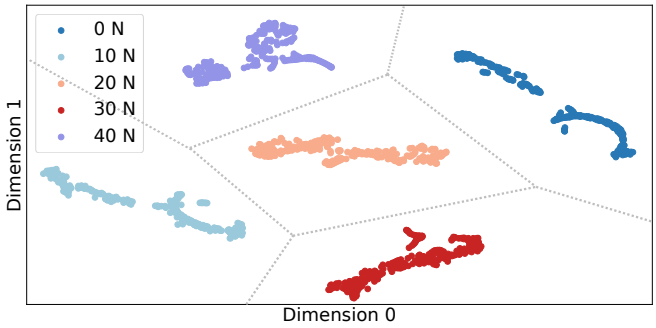

Fig. 5: The t-SNE visualization of learned latent representation. In different trials, the robot is subjected to a constant backward force of different magnitudes during forward movement.

trained with privileged force information are more responsive to unexpected perturbations. Additionally, the decrease in LO and RT values for PA-LOCO compared to MEFI further validates the efficacy of the residual network in PA-LOCO.

C. Analysis of the latent representation

To gain deeper insights into the role of the encoder in discerning perturbations of varying magnitudes and directions, a series of simulation experiments are conducted. The first experiment involves a robot moving on flat terrain at a forward speed of 0.5 m/s while experiencing a constant backward force throughout the process. Across multiple trials, consistent external forces of varying magnitudes but uniform direction are applied, specifically 0 N, 10 N, 20 N, 30 N, and 40 N, respectively. Analysis based on the t-distributed stochastic neighbor embedding (t-SNE) plot, as depicted in Fig. 5 reveals the distinct distribution of latent features generated by multi-encoder for different scenarios in the latent space. The analysis indicates that the learned policy is capable of discerning external forces of varying magnitudes, all in the same direction.

To further investigate the ability of a multi-encoder structure to discern consistent forces from different directions, the robot is given identical velocity commands on flat terrain while consistently experiencing external forces of 20 N from four directions: forward, backward, left, and right. The correlation heatmap between the force directions and each

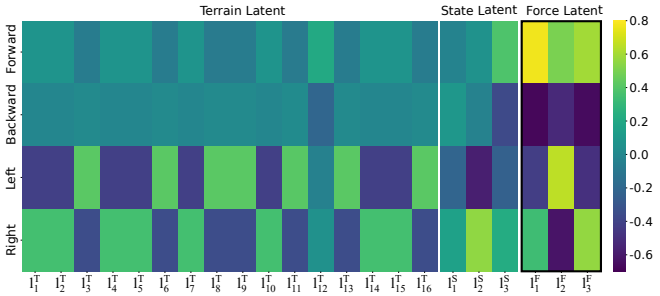

Fig. 6: The correlation heatmap between latent features and different force directions. l^T, l^S, l^F denote the latent features of the terrain profile T_t, robot state S_t, and external forces F_t respectively. In different trials, the robot experiences a constant external force of 20 N in different directions.

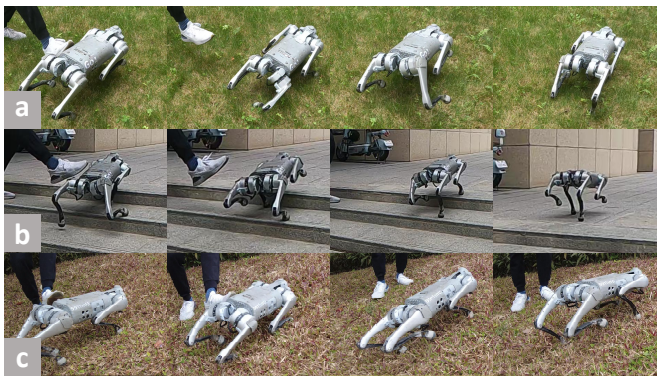

Fig. 7: Outdoor experiments. (a) Grass. (b) Stairs. (c) Slope.

component of the latent feature is shown in Fig. 6. It can be observed that considering each component of the latent feature of the external force l^F, there are four different forms of correlation between different directions and latent features of external forces. Note that the presence of forward forces is positively correlated with the three components of l^F, while backward forces correlate negatively. Similarly, for each component of l^F, different lateral forces show opposite correlations. Furthermore, some components of the state latent feature l^S show a slightly higher correlation with the direction of the external forces, due to variations in linear trunk velocity and foot-end contact state, similar to the robot's response to a frontal kick to a frontal kick illustrated in Session *A*. It implies that the multi-encoder structure enables the robot to distinguish between various types of privileged information, allowing it to make decisions based on the latent features.

D. Locomotion control in various outdoor environments

As shown in Fig. 7, the proposed algorithm is tested in various outdoor environments with velocity commands ranging from $[0,1]$ m/s via joystick. The robot demonstrates robustness to unexpected lateral kicks across different terrains, including grass, stairs, and slopes. Experiments confirm that the robot quickly recovers its original state after impacts. Notably, when ascending a 20-degree slope (Figure 6(c)), the robot's trunk height significantly decreases after impact to maintain balance.

VI. Conclusions

This paper presented a new teacher-student architecture featuring a residual policy and multi-encoder structure for robust quadruped locomotion. The residual policy could mitigate performance degradation during policy transfer. At the same time, the multi-encoder structure could decouple the latent features of external perturbations from those of other information, improving robot robustness. Physical experiments confirmed the controller's ability to tolerate unexpected perturbations across various terrains. Simulations demonstrated the multi-encoder's capacity to discern external forces of different magnitudes and directions. However, the current design requires tuning the importance of multiple encoder outputs for varying scenarios. Therefore, one future work will incorporate an attention mechanism to balance the encoder's outputs efficiently.

References

[1] J. Tan, T. Zhang, E. Coumans, A. Iscen, Y. Bai, D. Hafner, S. Bohez, and V. Vanhoucke, "Sim-to-real: Learning agile locomotion for quadruped robots," in *Proc. Robot. Sci. Syst.*, Pittsburgh, Pennsylvania, Jun. 2018.

[2] J. Hwangbo, J. Lee, A. Dosovitskiy, D. Bellicoso, V. Tsounis, V. Koltun, and M. Hutter, "Learning agile and dynamic motor skills for legged robots," *Sci. Robot.*, Jan. 2019.

[3] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Sci. Robot.*, Oct. 2020.

[4] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Sim-to-real transfer of robotic control with dynamics randomization," in *2018 IEEE Int. Conf. Robot. Autom. ICRA*. Brisbane, QLD: IEEE, May 2018, pp. 3803–3810.

[5] A. Kumar, Z. Fu, D. Pathak, and J. Malik, "RMA: Rapid motor adaptation for legged robots," in *Proc. Robot. Sci. Syst.*, Virtual, Jul. 2021.

[6] T. Miki, J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning robust perceptive locomotion for quadrupedal robots in the wild," *Sci. Robot.*, Jan. 2022.

[7] Y. Kim, H. Oh, J. Lee, J. Choi, G. Ji, M. Jung, D. Youm, and J. Hwangbo, "Not only rewards but also constraints: Applications on legged robot locomotion," *IEEE Trans. Robot.*, vol. 40, pp. 2984–3003, 2024.

[8] N. Rudin, D. Hoeller, P. Reist, and M. Hutter, "Learning to walk in minutes using massively parallel deep reinforcement learning," in *Proc. 5th Conf. Robot Learn.* PMLR, Jan. 2022, pp. 91–100.

[9] 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 Robot. Autom. Lett.*, vol. 7, no. 2, pp. 4630–4637, Apr. 2022.

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

[11] C. Yang, K. Yuan, Q. Zhu, W. Yu, and Z. Li, "Multi-expert learning of adaptive legged locomotion," *Sci. Robot.*, vol. 5, no. 49, p. eabb2174, Dec. 2020.

[12] G. B. Margolis, T. Chen, K. Paigwar, X. Fu, D. Kim, S. bae Kim, and P. Agrawal, "Learning to jump from pixels," in *Proc. 5th Conf. Robot Learn.* PMLR, Jan. 2022, pp. 1025–1034.

[13] N. Rudin, H. Kolvenbach, V. Tsounis, and M. Hutter, "Cat-like jumping and landing of legged robots in low gravity using deep reinforcement learning," *IEEE Trans. Robot.*, vol. 38, no. 1, pp. 317–328, Feb. 2022.

[14] Z. Zhuang, Z. Fu, J. Wang, C. G. Atkeson, S. Schwertfeger, C. Finn, and H. Zhao, "Robot parkour learning," in *Proc. 7th Conf. Robot Learn.* PMLR, Dec. 2023, pp. 73–92.

[15] F. Shi, T. Homberger, J. Lee, T. Miki, M. Zhao, F. Farshidian, K. Okada, M. Inaba, and M. Hutter, "Circus anymal: A quadruped learning dexterous manipulation with its limbs," in *2021 IEEE Int. Conf. Robot. Autom. ICRA*, May 2021, pp. 2316–2323.

[16] X. Cheng, A. Kumar, and D. Pathak, "Legs as manipulator: Pushing quadrupedal agility beyond locomotion," in *2023 IEEE Int. Conf. Robot. Autom. ICRA*, May 2023, pp. 5106–5112.

[17] Y. Ji, G. B. Margolis, and P. Agrawal, "DribbleBot: Dynamic legged manipulation in the wild," in *2023 IEEE Int. Conf. Robot. Autom. ICRA*, May 2023, pp. 5155–5162.

[18] X. B. Peng, E. Coumans, T. Zhang, T.-W. E. Lee, J. Tan, and S. Levine, "Learning agile robotic locomotion skills by imitating animals," in *Proc. Robot. Sci. Syst.*, Jul. 2020.

[19] G. B. Margolis, G. Yang, K. Paigwar, T. Chen, and P. Agrawal, "Rapid locomotion via reinforcement learning," *Int. J. Robot. Res.*, Jan. 2024.

[20] Z. Luo, E. Xiao, and P. Lu, "FT-net: Learning failure recovery and fault-tolerant locomotion for quadruped robots," *IEEE Robot. Autom. Lett.*, vol. 8, no. 12, pp. 8414–8421, Dec. 2023.

[21] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017. [Online]. Available: https://arxiv.org/abs/1707.06347

[22] X. Zhang, Z. Xiao, Q. Zhang, and W. Pan, "SYNLOCO: Synthesizing central pattern generator and reinforcement learning for quadruped locomotion," 2023. [Online]. Available: https://arxiv.org/abs/2310.06606

[23] Makoviychuk *et al.*, "Isaac gym: High performance GPU based physics simulation for robot learning," in *Proc. Neural Inf. Process. Syst. Track Datasets Benchmarks*, vol. 1, Dec. 2021.