Risk-Specific Multi-Expert Distillation for Autonomous Robots: The Case of Quadrupedal Handstand Recovery

Camille F. Buschman*,1, Kousheek Chakraborty*,1, Abeje Yenehun Mersha[1]

***Abstract*—Autonomous robots operate under various forms of uncertainty, disturbance, and unexpected interaction with their environment, all of which can introduce risk during decision-making and control. In this work, we introduce a structured risk taxonomy that distinguishes between risk indicators, risk measures, and undesired outcomes, and organises risk indicators along internal/external and implicit/explicit axes. The taxonomy is intended as a general framework for reasoning about risk in autonomous robotic systems. As a demonstrative use case, we develop a risk-specific multi-expert distillation approach for quadrupedal handstand recovery, training one expert policy on one representative risk example per category: joint friction variation (internal-implicit), joint failure (internal-explicit), external perturbation forces (external-implicit), and static obstacles in the landing environment (external-explicit). These experts are then distilled into a single context-agnostic student policy. We evaluate the approach in simulation across perturbations associated with the proposed risk categories and demonstrate transfer to the Spot quadruped in a handstand landing task. The preliminary results in simulation show that risk-specific experts have a success rate of over 90% in handling their risk category and they can be combined into a unified policy with a success rate of over 85% over all risk categories. Although the policy was successfully deployed on hardware, the hardware performance did not fully match the simulation results. In particular, post-landing instability indicates remaining sim-to-real gaps, likely caused by unmodeled dynamics and contact effects. Future work will therefore focus on improving sim-to-real transfer and integrating onboard perception for obstacle-aware recovery to evaluate the obstacle-aware landing behaviours within the external-explicit risk category.**

***Index Terms*—reinforcement learning, multi-expert distillation, risk-aware robotics**

I. Introduction

Risk is inherent to the environments in which autonomous robots operate [1]. A robot that moves through the world never acts with complete certainty. For instance, its sensors are noisy, its actuators are imperfect, its body state evolves dynamically, and the environment may change in ways that are only partially observable. For humans, such uncertainty is not only a limitation, but also a source of information. We slow down when we are unsure, adapt when our body feels unstable, and act more cautiously when the situation becomes difficult to interpret.

Video: https://youtu.be/q2nxxEwOlSw

* Equal contribution.
[1]Smart Mechatronics and Robotics research group, Saxion University of Applied Science, Enschede, The Netherlands.
Corresponding author: Camille Buschman c.f.buschman@saxion.nl

Fig. 1. Overview of the handstand recovery task on LUCI, a Boston Dynamics Spot at Saxion SMART Mechatronics and Robotics Research Group. The robot transitions from a handstand to a four-legged stance using the proposed policy. For clarity, the visualization uses offset frames to reduce overlap between consecutive poses.

Robotic systems, in contrast, often address risk reactively through predefined safety rules, margins, or constraints [2]. Although these mechanisms are useful for enforcing safe behaviour, they can become overly conservative or difficult to apply in dynamic environments where future system and agent behaviour is uncertain [3]. Reinforcement learning (RL) [4] offers an alternative to rule-based approaches by enabling agents to learn adaptive behaviour through interaction.

Recent advances in multi-expert learning [5] and policy distillation [6] suggest that complex behaviour can be learned by training specialized expert policies for specific tasks. Inspired by this idea, we investigate whether risk-aware behaviour can be learned by training separate RL agents for distinct categories of risk indicators. Rather than training a single policy to handle all forms of uncertainty at once, each agent is exposed to one specific risk category and learns to mitigate the corresponding undesired outcomes. This decomposition allows each expert to develop dedicated strategies for its category, rather than averaging over the competing failure modes that a single unified policy would have to accommodate at once.

In this work, we introduce a risk framework that distinguishes between four categories of risk indicators: internal implicit, internal explicit, external implicit, and external explicit. This separation matters because each category calls for a different approach: internal risks must be inferred from

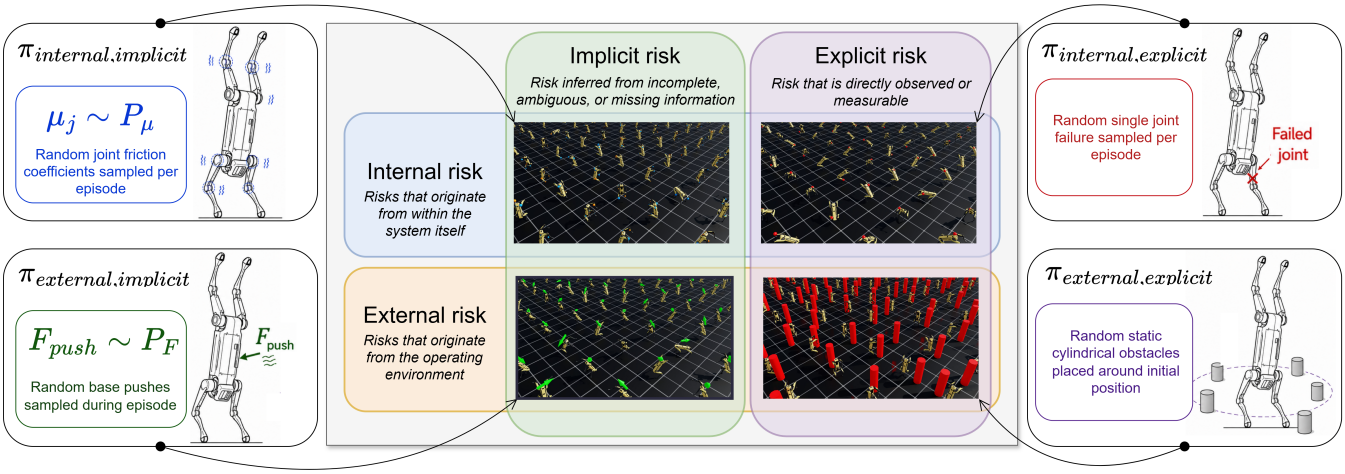

Fig. 2. Risk taxonomy organizing risk indicators in a framework along two dimensions: origin (internal vs. external) and form (explicit vs. implicit). The four resulting categories define the specialist training environments used for expert policy learning. The risks mentioned in each category are representative examples they can be extended to include more instances per category.

proprioceptive cues, external risks engage exteroceptive perception. Implicit risks call for continuous adaptation, while explicit risks require discrete responses to identifiable events. These categories structure uncertainty according to where it originates and how it manifests. We instantiate the categories in four RL training environments and train one agent per category.

As a demonstrative use case, we apply this framework to a quadruped robot recovering from a handstand configuration to a stable four-legged stance as show in Figure 1. The objective is to successfully complete the landing while mitigating risks related to joint failure, unsafe impact, actuator degradation, and environmental disturbances. Through this experiment, we explore how structured risk indicators can be used to train specialized risk-aware expert policies, forming a first step toward multi-expert learning and policy distillation for adaptive robotic risk mitigation.

The main contributions of this work:

- A structured taxonomy of robotic risk indicators based on their origin and form.
- A risk-specific multi-expert distillation framework that trains one specialist policy per category and distils them into a single context-agnostic student.
- A hardware demonstration on a Boston Dynamics Spot quadruped for handstand recovery.

II. Risk taxonomy

Following prior work on risk-aware decision making in robotics, we treat risk as a representation of how uncertainty may affect future system behaviour and mission objectives [7]. More generally, this framing is grounded in the ISO definition of risk as the effect of uncertainty on objectives [8].

In this work, we distinguish between three related but different concepts: risk indicators, risk measures, and undesired outcomes. This distinction is introduced as a working conceptualization for risk-aware robotic decision-making.

- **A risk indicator** is an observed or inferred signal relevant for the task at hand, that suggests the presence or increase of uncertainty in the robotic system or its environment. Such uncertainty may arise from incomplete or noisy information, including sensor noise, model inaccuracies, actuation uncertainty, or partially observed environmental conditions [1].
- **A risk measure** represents the estimated level of risk associated with uncertain outcomes. Following [7], risk measures can be interpreted as mappings from uncertain outcomes to scalar quantities. In this work, we consider such measures to be derived from one or more risk indicators.
- **An undesired outcome** is the event associated with this estimated risk that the robot aims to avoid, such as collision, loss of balance, unsafe control behaviour, or task failure.

This distinction makes it possible to treat risk as an anticipatory signal. Risk indicators, interpreted through corresponding risk measures, can signal that the robot is moving toward a potentially unsafe future state before an undesired outcome has occured. These risk measures can then be used to adapt the robot's behaviour in advance.

We further extend this idea by categorizing risk indicators along two dimensions: their origin, and their form as depicted in Figure 2. The origin of a risk indicator can be internal, when it relates to the robotic system itself, or external, when it relates to the environment. The form of an indicator can be explicit, when it is directly observable or measurable, or implicit, when it is inferred from incomplete, ambiguous, or missing information.

Combining these dimensions results in four categories of risk indicators:

1) Internal Implicit risk
2) Internal Explicit risk
3) External Implicit risk
4) External Explicit risk

To the best of our knowledge, this work is among the first in robotics to explicitly structure risk indicators along both internal/external and explicit/implicit dimensions.

III. Problem statement

The objective of this work is to use the proposed framework to enable a quadruped robot to safely recover from a handstand configuration to a stable four-legged stance. In our use case, the robot is already in the handstand state, and therefore we focus on the landing phase.

We model this task as a context-conditioned Markov Decision Process [9] $\mathcal{M} = (\mathcal{X}, \mathcal{A}, \mathcal{C}, \mathcal{P}, r, \gamma, \rho_0)$ with continuous state space $\mathcal{X}$, continuous action space $\mathcal{A}$, reward $r : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$, discount $\gamma \in [0, 1)$, and initial distribution ρ_0 over handstand poses. The context space $\mathcal{C} = \mathcal{O} \times \mathcal{F}$ corresponds to the four categories of the risk taxonomy, with $\mathcal{O} = \{\text{internal}, \text{external}\}$ and $\mathcal{F} = \{\text{implicit}, \text{explicit}\}$. The transition kernel

$$\mathcal{P} : \mathcal{X} \times \mathcal{A} \times \mathcal{C} \;\to\; \Delta(\mathcal{X}) \tag{1}$$

is parametrised by the active context $c \in \mathcal{C}$: for each c, the perturbation governing $\mathcal{P}(\cdot \mid x, a, c)$ is sampled from a risk-specific distribution $\mathcal{R}_c$. The context is fixed at the start of each episode and is not observed by the policy.

We then seek a single context-agnostic policy $\pi : \mathcal{X} \to \Delta(\mathcal{A})$ that maximises the expected discounted return over the context distribution,

$$J(\pi) \;=\; \mathbb{E}_{c \sim \mathcal{U}(\mathcal{C})}\, \mathbb{E}_{\pi,\, \mathcal{P}(\cdot \mid \cdot, \cdot,\, c)} \left[\sum_{t=0}^{T} \gamma^t \, r(x_t, a_t) \right] \tag{2}$$

IV. Approach

We frame safe handstand termination as a two-stage multi-expert distillation problem similar to [5], [6]. In the first stage, we train one specialist RL policy $\pi_{o,f}$ per category of the risk framework as seen in Figure 2, where $o \in \mathcal{O}$ denotes the risk origin and $f \in \mathcal{F}$ its form. Each expert is optimised on a distinct training distribution $\mathcal{R}_{o,f}$ that isolates a single risk source, while the observation space, action space, reward function, network architecture, and initial state distribution are held identical across all four. In the second stage, the frozen experts are distilled into a single category-agnostic student policy π_θ intended for deployment. Episodes are initialised from a handstand configuration sampled from the terminal state distribution of a pre-trained handstand policy, with some randomisation over the base orientation and joint angles; the goal of the policy is to return the robot to a stable quadrupedal stance without incurring damage to itself or its surroundings.

a) Risk specific environments: Each expert $\pi_{o,f}$ is trained under perturbations restricted to its category in the framework as shown in Figure 2.

1) $\pi_{\text{internal,implicit}}$ sees randomized joint friction sampled per episode.
2) $\pi_{\text{internal,explicit}}$ encounters a single joint failure on a uniformly sampled joint per episode.
3) $\pi_{\text{external,implicit}}$ experiences a push applied at the base at random intervals within the episode.
4) $\pi_{\text{external,explicit}}$ is trained in environments containing static cylindrical obstacles placed uniformly around the robot's initial position.

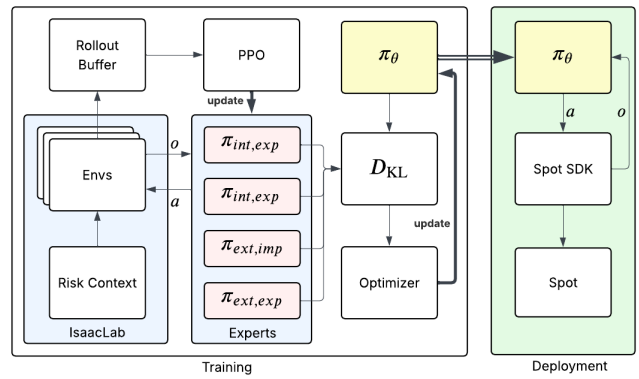

Fig. 3. Two-stage training pipeline. In Stage 1, four specialist policies are trained in risk-specific environments. In Stage 2, the frozen experts supervise a single context-agnostic student policy. Finally the student policy is deployed on hardware.

The risks introduced to each policy are representative examples of a risk within that category. However, the framework could be extended to include more instances per risk category.

b) Observations: The policy receives an observation $\mathbf{o}_t \in \mathbb{R}^{45}$ at each control step:

$$\mathbf{o}_t \;=\; \left[\, \mathbf{v}_t^b,\; \boldsymbol{\omega}_t^b,\; \mathbf{g}_t^b,\; \mathbf{q}_t,\; \dot{\mathbf{q}}_t,\; \mathbf{a}_{t-1} \,\right]^\top \tag{3}$$

with:

- $\mathbf{v}_t^b$ is the base linear velocity in the body frame,
- $\boldsymbol{\omega}_t^b$ is the base angular velocity in the body frame,
- $\mathbf{g}_t^b$ is the gravity vector projected into the body frame,
- $\mathbf{q}_t$ are the joint positions,
- $\dot{\mathbf{q}}_t$ are the joint velocities,
- $\mathbf{a}_{t-1}$ are the previous policy output.

c) Action: The policy outputs a vector $\mathbf{a}_t \in \mathbb{R}^{12}$ of desired joint position offsets relative to the nominal stance defining per-joint targets at each timestep t

$$\mathbf{q}_t^{\text{des}} \;=\; \mathbf{q}_0 \;+\; \mathbf{a}_t \tag{4}$$

These targets are tracked by a low-level joint-space PD controller that produces joint torques for each actuator of the robot

$$\boldsymbol{\tau}_t \;=\; K_p \left(\mathbf{q}_t^{\text{des}} - \mathbf{q}_t \right) \;-\; K_d \, \dot{\mathbf{q}}_t \tag{5}$$

with fixed stiffness K_p and damping K_d. Torques are clipped to actuator limits before being applied to the simulator.

d) Rewards: The reward combines a posture-tracking task signal with safety and regularisation penalties:

$$r \;=\; r_{\text{stance}} + r_{\text{impact}} + r_{\text{act}} + r_{\text{term}} \tag{6}$$

with

$$\begin{aligned} r_{\text{stance}} &= -\, w_1 \left\| \mathbf{q} - \mathbf{q}_0 \right\|_2^2 \\ r_{\text{impact}} &= -\, w_2 \sum_{i=1}^{4} \left\| \mathbf{F}_i \right\|_2^2 \\ r_{\text{act}} &= -\, w_3 \left\| \mathbf{a}_t - \mathbf{a}_{t-1} \right\|_2^2 \\ r_{\text{term}} &= -\, w_4 \, [\text{terminated}] \end{aligned}$$

where $\mathbf{q}_0$ is the default standing configuration and $\mathbf{F}_i$ is the contact force on foot i. While training, the following weights were used $w_1 = 1, w_2 = 10, w_3 = 1e-3, w_4 = 500$.

e) Terminations: An episode terminates when any of the following conditions are encountered:

1) *Loss of support:* Any of the front two feet leaves the ground.
2) *Body collision:* Any non-foot link contacting the ground, or any link contacting an obstacle.

f) Distillation: Once the four experts have converged, we freeze their weights and distil them into a single student policy π_θ via on-policy imitation [10]. At each training step, a risk category (o, f) is sampled uniformly, the corresponding environment is instantiated under $\mathcal{R}_{o,f}$, and the student is rolled out to visit states from its own action distribution; the active expert $\pi_{o,f}$ then provides supervision over those visited states. The student receives only the standard proprioceptive observation $\mathbf{o}_t$ and is given no indication of which risk category is active, it must implicitly infer the relevant risk context. Training minimises the expected Kullback–Leibler divergence between the student's and active expert's action distributions over student-induced rollouts:

$$\mathcal{L}(\theta) = \mathbb{E}\big[\, D_{\mathrm{KL}}\big(\pi_\theta(\cdot|\mathbf{o}_t) \,\|\, \pi_{o,f}(\cdot|\mathbf{o}_t)\big)\big] \tag{7}$$

resulting in a single policy capable of safe landing across all risk categories at deployment.

V. Preliminary Results

To train our RL policies, we developed a parallelized simulation environment similar to [11] using Isaac Lab [12], a robotics learning framework built on top of Isaac Sim. For policy optimization, we used the Proximal Policy Optimization (PPO) [13] algorithm from the RSL-RL library [14]. The training ran 4096 environments in parallel for 1000 policy iterations with wall-clock training time averaging 15–20 minutes. The training was conducted on a workstation with the following specifications: Intel Core i9, NVIDIA RTX 5000 ADA, 32 GB RAM. Once the training was completed, the model was copied onto an NVIDIA Jetson Nano mounted on the Boston Dynamics Spot which ran the inference pipeline based on the work from [15] and communicated with the Spot via the provided SDK. An overview on the training and deployment pipeline can be seen in Figure 3.

We first conducted a cross-evaluation in simulation in which each expert policy $\pi_{o,f}$ was rolled out for 1000 episodes in every risk environment $\mathcal{R}_{o',f'}$, covering both matched and mismatched risk categories. We report the success rate, defined as the fraction of episodes in which the robot reaches a stable four-legged stance without early termination, in Table I. Each expert achieves the highest success rate on its own training distribution but degrades when evaluated outside it. The distilled student π_θ achieves good performance across all risk categories.

We then deployed the distilled student on a physical Boston Dynamics Spot as seen in Figure 1. A separately trained handstand policy first brings the robot into the inverted configuration; upon receiving a "land" command from the operator, control hands off to π_θ, which runs the joint level controller that returns the robot to a four-legged stance. However, we noticed that the robot collapses once all 4 feet touch the ground.

TABLE I
Cross-evaluation success rates averaged over 1000 episodes.

Test env.	$\pi_{\mathsf{int,imp}}$	$\pi_{\mathsf{int,exp}}$	$\pi_{\mathsf{ext,imp}}$	$\pi_{\mathsf{ext,exp}}$	π_θ
$\mathcal{R}_{\mathsf{int,imp}}$	**0.94**	0.42	0.38	0.21	**0.91**
$\mathcal{R}_{\mathsf{int,exp}}$	0.25	**0.91**	0.29	0.18	**0.87**
$\mathcal{R}_{\mathsf{ext,imp}}$	0.11	0.45	**0.93**	0.16	**0.89**
$\mathcal{R}_{\mathsf{ext,exp}}$	0.08	0.52	0.64	**0.90**	**0.86**
Mean	0.35	0.58	0.56	0.36	**0.88**

VI. Conclusion

We presented a structured risk taxonomy and a risk-specific multi-expert distillation framework for quadrupedal handstand recovery. By training specialized expert policies for distinct risk categories and distilling them into a single context-agnostic student, we demonstrated good performance across the different perturbations related to the risk categories in simulation as well as a successful deployment on hardware.

In simulation, each expert achieved a success rate above 90% on its own training distribution but generalised poorly to other risk categories. The distilled policy recovered comparable performance with a success rate over 85% across all four risk categories despite operating without knowledge of the active risk.

While the policy transferred to the physical platform, we observed sim-to-real discrepancies. In particular, the robot collapsed after landing, potentially indicating unmodeled dynamics and contact effects that were not captured during training. Identifying this sim-to-real gap is a direction for future work.

Additionally, current hardware experiments do not yet incorporate exteroceptive perception. A next step is to develop an onboard perception pipeline on Spot to detect obstacles using its sensors to test the obstacle-aware landing behaviours corresponding to the external-explicit risk category.

References

[1] J. Bongard, "Probabilistic Robotics. Sebastian Thrun, Wolfram Burgard, and Dieter Fox. (2005, MIT Press.) 647 pages," *Artificial Life*, vol. 14, no. 2, pp. 227–229, Apr. 2008, _eprint: https://direct.mit.edu/artl/article-pdf/14/2/227/1662501/artl.2008.14.2.227.pdf, ISSN: 1064-5462. DOI: 10.1162/artl.2008.14.2.227. [Online]. Available: https://doi.org/10.1162/artl.2008.14.2.227.

[2] L. Brunke et al., "Safe learning in robotics: From learning-based control to safe reinforcement learning," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 5, no. Volume 5, 2022, pp. 411–444, 2022, ISSN: 2573-5144. DOI: https://doi.org/10.1146/annurev-control-042920-020211. [Online]. Available: https://www.annualreviews.org/content/journals/10.1146/annurev-control-042920-020211.

[3] H.-J. Huang, K.-C. Huang, M. Čáp, Y. Zhao, Y. N. Wu, and C. L. Baker, "Planning on a (risk) budget: Safe non-conservative planning in probabilistic dynamic environments," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 10 257–10 263. DOI: 10.1109/ICRA48506.2021.9561745.

[4] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: A Bradford Book, 2018, ISBN: 0262039249.

[5] N. Rudin, J. He, J. Aurand, and M. Hutter, *Parkour in the Wild: Learning a General and Extensible Agile Locomotion Policy Using Multi-expert Distillation and RL Fine-tuning*, arXiv:2505.11164 [cs], May 2025. DOI: 10.48550/arXiv.2505.11164. [Online]. Available: http://arxiv.org/abs/2505.11164.

[6] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Science Robotics*, vol. 5, no. 47, eabc5986, Oct. 2020. DOI: 10.1126/scirobotics.abc5986. Accessed: May 14, 2026. [Online]. Available: https://www.science.org/doi/10.1126/scirobotics.abc5986.

[7] A. Majumdar and M. Pavone, "How should a robot assess risk? towards an axiomatic theory of risk in robotics," in *Robotics Research*, N. M. Amato, G. Hager, S. Thomas, and M. Torres-Torriti, Eds., Cham: Springer International Publishing, 2020, pp. 75–84, ISBN: 978-3-030-28619-4.

[8] *Iso 31000:2018 risk management — guidelines*, Geneva, Switzerland: International Organization for Standardization, 2018. [Online]. Available: https://www.iso.org/standard/65694.html.

[9] A. Hallak, D. D. Castro, and S. Mannor, *Contextual Markov Decision Processes*, arXiv:1502.02259 [stat], Feb. 2015. DOI: 10.48550/arXiv.1502.02259. [Online]. Available: http://arxiv.org/abs/1502.02259.

[10] S. Ross, G. Gordon, and D. Bagnell, "A reduction of imitation learning and structured prediction to no-regret online learning," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, G. Gordon, D. Dunson, and M. Dudík, Eds., ser. Proceedings of Machine Learning Research, vol. 15, Fort Lauderdale, FL, USA: PMLR, Nov. 2011, pp. 627–635. [Online]. Available: https://proceedings.mlr.press/v15/ross11a.html.

[11] N. Rudin, D. Hoeller, P. Reist, and M. Hutter, "Learning to walk in minutes using massively parallel deep reinforcement learning," in *Proceedings of the 5th Conference on Robot Learning*, A. Faust, D. Hsu, and G. Neumann, Eds., ser. Proceedings of Machine Learning Research, vol. 164, PMLR, Aug. 2022, pp. 91–100. [Online]. Available: https://proceedings.mlr.press/v164/rudin22a.html.

[12] M. Mittal et al., "Isaac lab: A gpu-accelerated simulation framework for multi-modal robot learning," *arXiv preprint arXiv:2511.04831*, 2025. [Online]. Available: https://arxiv.org/abs/2511.04831.

[13] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, *Proximal Policy Optimization Algorithms*, arXiv:1707.06347 [cs.LG], Aug. 2017. DOI: 10.48550/arXiv.1707.06347. [Online]. Available: http://arxiv.org/abs/1707.06347.

[14] C. Schwarke, M. Mittal, N. Rudin, D. Hoeller, and M. Hutter, *RSL-RL: A Learning Library for Robotics Research*, arXiv:2509.10771 [cs], Sep. 2025. DOI: 10.48550/arXiv.2509.10771. [Online]. Available: http://arxiv.org/abs/2509.10771.

[15] A. J. Miller, F. Yu, M. Brauckmann, and F. Farshidian, *High-Performance Reinforcement Learning on Spot: Optimizing Simulation Parameters with Distributional Measures*, arXiv:2504.17857 [cs.LG], Jul. 2025. DOI: 10.48550/arXiv.2504.17857. Accessed: May 14, 2026. [Online]. Available: http://arxiv.org/abs/2504.17857.