EmoLo: Emotion-Inspired Expressive Locomotion via Single-Policy Reinforcement Learning on Low-Cost Bipedal Robots

Masato Kobayashi[1,2*]

***Abstract*— Legged robots in human-centered settings should combine reliable locomotion with behavior that is expressive and easy to interpret. This paper presents a style-conditioned reinforcement learning framework for Open Duck Mini V2 that generates emotion-inspired walking behavior through three discrete styles—*Happy*, *Neutral*, and *Sad*—using a single shared policy. The policy is trained in simulation with a two-part objective: locomotion terms are inherited from an open-source baseline, while a compact *style* objective modulates head-pitch posture and motion activity through bounded rewards and a command-dependent gate. Training incorporates sim-to-real considerations such as sensor noise, delay, external pushes, and motor limits, and the controller is exported to ONNX for onboard inference. We evaluate the method in simulation and on hardware under a controlled forward-walking protocol, focusing on trial completion and head-pitch trajectories. The results show successful locomotion with consistent style-dependent head behavior, demonstrating that emotion-inspired expressive modulation can be integrated into a deployable low-cost bipedal controller without multiple policies. Additional material is available at `https://mertcookimg.github.io/emolo`**

I. INTRODUCTION

Legged robots are increasingly being deployed in human-facing environments, where not only task performance but also motion quality influences how their behavior is perceived. In such settings, differences in posture, timing, and activity can make robot motion appear calm, lively, subdued, or otherwise expressive. This has motivated growing interest in locomotion systems that remain physically reliable while also producing motion that is visually distinct and easy to interpret [1]–[5].

Achieving such expressive behavior on real robots is challenging. Unlike purely graphical characters, physical robots must satisfy contact dynamics, actuator limits, sensing constraints, and real-time control requirements. As a result, motion *believability* can easily be degraded by jitter, impact noise, or unnatural timing, especially in human-facing applications [6]. These constraints suggest that expressive modulation should be developed together with low-level control and sim-to-real robustness, rather than treated only as an animation problem.

Existing approaches address this challenge from two main directions. On one side, animation-driven and character-oriented systems can produce visually compelling behaviors, but often rely on complex architectures involving motion authoring, blending, policy switching, or external runtime modules [7], [8]. On the other side, reinforcement learning (RL) has achieved strong results for legged locomotion across a wide range of platforms, including open-source and relatively low-cost robots [9]–[13]. These controllers are effective for stability, tracking, and sim-to-real transfer, but they are typically task-oriented and often yield visually homogeneous motion.

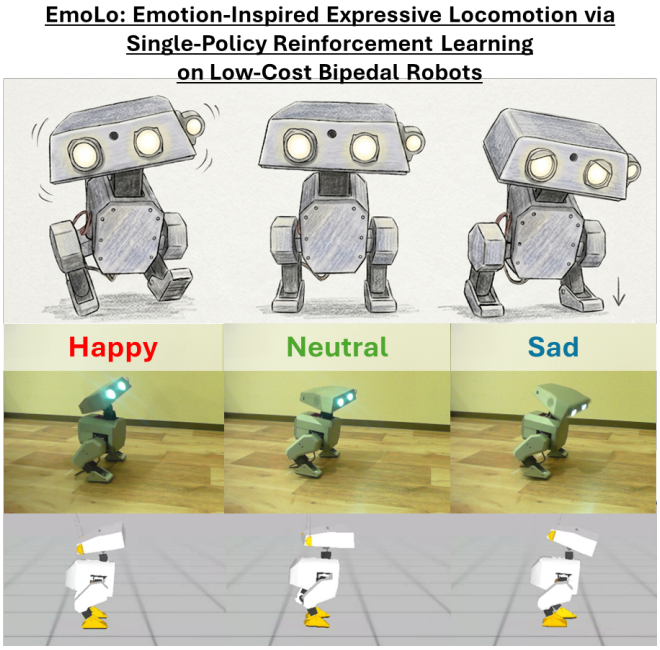

Fig. 1. Concept illustration of the three emotion-inspired style conditions (*Happy*, *Neutral*, and *Sad*) across artwork, hardware, and simulation, highlighting the head-pitch-related behavior targeted in this work.

This contrast highlights a practical gap: how to introduce expressive and interpretable motion variation into RL-based locomotion without sacrificing simplicity, deployability, or real-world robustness.

In this paper, we propose a style-conditioned reinforcement learning framework that enables emotion-inspired expressive locomotion within a single deployable policy. We focus on Open Duck Mini V2 [13], a compact open-source bipedal platform, and introduce a lightweight reward-level design that modulates head-pitch posture and motion activity across three styles—*Happy*, *Neutral*, and *Sad*. The locomotion component is inherited from an open-source baseline to preserve stability and sim-to-real compatibility, while the added style component produces behaviorally distinct motion without requiring multiple policies or external animation modules.

The University of Osaka, [1] D3 Center, The University of Osaka, [2] Graduate School of Maritime Sciences, Kobe University, * corresponding author: kobayashi.masato.cmc@osaka-u.ac.jp

We evaluate the proposed method in simulation and on hardware under a controlled walking protocol, varying only the style label while monitoring trial completion and head-pitch trajectories. The results show successful forward locomotion together with consistent style-dependent differences in head behavior, supporting the feasibility of lightweight expressive modulation on a deployable low-cost bipedal controller.

The main contributions of this paper are as follows:

- **Emotion-inspired expressive locomotion on a low-cost biped:** Demonstration that visually distinct walking behaviors can be generated on a compact, open-source bipedal robot under practical deployment constraints.
- **Single-policy style-conditioned RL design:** A lightweight framework that introduces interpretable style modulation through reward-level conditioning while preserving the underlying locomotion controller.
- **Practical sim-to-real deployment:** Direct transfer of the trained controller to real hardware via ONNX without additional system components or retraining.

The remainder of this paper is organized as follows. Section II reviews related work. Section III presents the proposed method. Section IV reports experimental results in simulation and on hardware. Section V concludes the paper.

II. RELATED WORK

A. Legged Locomotion with Reinforcement Learning

Reinforcement learning (RL) has become a standard approach for legged locomotion, enabling robust feedback policies through large-scale simulation. Prior work has demonstrated agile locomotion, command tracking, and strong sim-to-real transfer by combining policy learning with domain randomization, actuator modeling, and carefully designed low-level control interfaces [9]–[12], [14]. Simulation tools such as MuJoCo have further made it practical to train such policies efficiently while reducing the cost and risk of hardware trial-and-error [15]–[17].

Recent work has also emphasized reproducibility and accessibility through open-source and relatively low-cost robotic platforms [13], [18]–[23]. These platforms are important for embodied AI and HRI research, but their limited onboard computation, sensing, and actuation bandwidth require controllers to remain lightweight at deployment time.

Despite this progress, most RL-based locomotion controllers remain primarily task-oriented, focusing on stability, tracking, efficiency, and robustness. As a result, they often produce functionally effective but visually homogeneous behaviors. EmoLo is positioned in this gap: rather than replacing the locomotion pipeline, it extends a deployable RL controller with lightweight, reward-level style modulation for emotion-inspired expressive behavior on a low-cost biped.

B. Expressive and Character-based Robot Motion

A complementary line of research has examined how robot motion influences human perception and how animation principles can be used to create more believable and expressive behavior. Early work on entertainment and social robots emphasized motion composition, blending, and behavior authoring for lifelike interaction [7], [8], [24], while studies in HRI and expressive robotics have shown that body motion, posture, and timing are central to how humans interpret robot intent and affect [3], [25].

More recent work has brought these ideas into legged robotics by combining animation references with physically grounded control. RL-based imitation learning enables policies to track reference motions while maintaining closed-loop robustness [12], [26], [27], and systems such as Animated Cassie demonstrate that dynamic robots can be made more relatable through stylized motion design [28].

Particularly relevant are recent character-robot pipelines that integrate animation authoring with RL-based control. Grandia *et al.* combine artist-authored motions, multiple policies, and runtime animation engines for expressive behavior [7], while *Olaf* extends this direction to a constrained physical character with additional considerations such as impact reduction and thermal-aware control [8]. These systems achieve compelling results but rely on relatively complex architectures involving policy switching, animation blending, and operator interaction.

EmoLo is complementary to these approaches. Rather than constructing a full animation-driven pipeline or relying on multiple specialized policies, it introduces emotion-inspired, style-conditioned locomotion within a single RL policy on a compact, open-source biped. Expressive variation is produced through a minimal set of interpretable reward terms, primarily modulating head-related behavior during walking while preserving the underlying locomotion controller.

III. EMOLO: EMOTION-INSPIRED EXPRESSIVE LOCOMOTION VIA SINGLE-POLICY REINFORCEMENT LEARNING ON LOW-COST BIPEDAL ROBOTS

A. Overview

The proposed formulation is intended as a lightweight extension to an existing locomotion controller, adding emotion-inspired style modulation without changing the overall deployment architecture. We learn a single style-conditioned locomotion policy for Open Duck Mini V2 in the joystick task. At each control step, the policy receives a state vector and outputs joint action commands. The same policy is used for all styles, while an explicit style code selects the target expressive mode. Formally, the policy is defined as

$$a_t = \pi_\theta(o_t, s), \tag{1}$$

where t is the discrete control step index, $\pi_\theta(\cdot)$ is the policy function parameterized by trainable weights θ, o_t is the observation at time t, $s \in \{0, 1, 2\}$ is the style label (*Neutral*, *Happy*, *Sad*), and a_t is the action vector. This design enables online style switching without loading multiple controllers.

B. Command and Observation Design

The command vector contains planar velocity and head-orientation targets:

$$c_t = \big[v_x,\ v_y,\ \omega_z,\ q_{\text{neck}},\ q_{\text{head-p}},\ q_{\text{head-y}},\ q_{\text{head-r}}\big]. \tag{2}$$

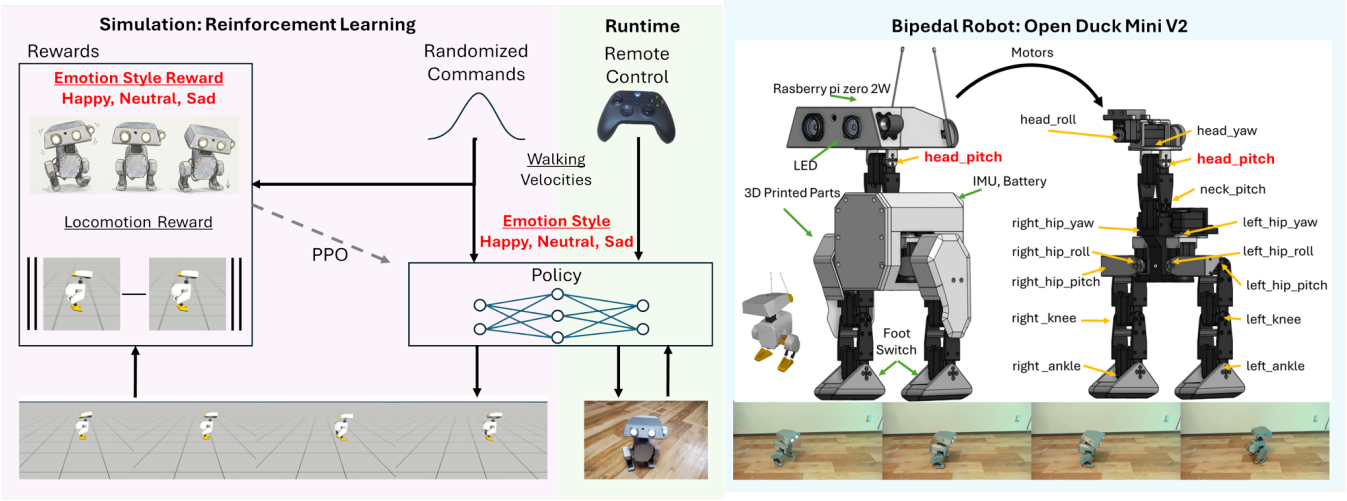

Fig. 2. EmoLo System Overview

Here, v_x and v_y are commanded linear velocities in the robot local frame, ω_z is the commanded yaw rate, and $q_{\text{neck}}, q_{\text{head-p}}, q_{\text{head-y}}, q_{\text{head-r}}$ are target neck pitch, head pitch, head yaw, and head roll, respectively. During training, c_t is randomly sampled from bounded ranges, with occasional zero-command episodes to improve standing behavior. In the inference setting used in this paper, the head- and neck-orientation commands are fixed to zero, i.e., $q_{\text{neck}} = q_{\text{head-p}} = q_{\text{head-y}} = q_{\text{head-r}} = 0$. Thus, the style-dependent differences reported in the experiments arise under identical head-related command inputs across styles.

The policy observation is a concatenation of proprioceptive and command-related features:

$$o_t = \big[\omega_t^{\text{imu}},\ a_t^{\text{imu}},\ c_t,\ \Delta q_t,\ \dot{q}_t,\ a_{t-1}, a_{t-2}, a_{t-3}, u_{t-1}, \kappa_t, \phi_t, e_s\big]. \tag{3}$$

Here, ω_t^{imu} and a_t^{imu} are IMU angular velocity and linear acceleration; Δq_t and $\dot{q}_t$ are joint position error and scaled joint velocity; $a_{t-1}, a_{t-2}, a_{t-3}$ are recent action-history terms; u_{t-1} is the previous motor target; κ_t denotes foot-contact states; $\phi_t = [\cos\varphi_t, \sin\varphi_t]$ is the imitation phase with phase angle φ_t; and e_s is a style one-hot vector (subscript s matches the discrete style label).

C. Actuation and Delay-Aware Control

The policy predicts normalized actions that are converted to motor targets by

$$u_t = u_0 + \alpha a_t, \tag{4}$$

where u_t is the motor target at step t, u_0 is the default actuator pose, and α is a scalar action scale. To match real-world constraints, we include random action delay (sampled from a short history buffer) and enforce motor speed limits:

$$u_t \leftarrow \text{clip}\big(u_t,\ u_{t-1} - \dot{u}_{\max}\Delta t,\ u_{t-1} + \dot{u}_{\max}\Delta t\big). \tag{5}$$

In this equation, $\dot{u}_{\max}$ is the maximum motor-target rate and Δt is the control interval. This prevents unrealistically fast target jumps and improves transfer robustness.

D. Reward Formulation

The per-step reward is designed as a simple sum of two roles:

$$r_t = r_t^{\text{loc}} + r_t^{\text{style}}, \tag{6}$$

where r_t is the total reward at step t, r_t^{loc} is the locomotion reward, and r_t^{style} is the style reward.

The locomotion part r_t^{loc} is directly reused from the OSS baseline [13]. Its role is to keep the controller physically reliable, independent of style. We implement it as a weighted sum of scalar reward channels; the coefficients w_{lv}–w_{ss} are listed in Table II and match the symbols below:

$$r_t^{\text{loc}} = w_{\text{lv}}\, r_t^{\text{lv}} + w_{\text{av}}\, r_t^{\text{av}} + w_{\text{al}}\, r_t^{\text{al}} + w_{\text{im}}\, r_t^{\text{im}} + w_\tau\, r_t^{\tau} + w_{\dot{a}}\, r_t^{\dot{a}} + w_{\text{ss}}\, r_t^{\text{ss}}, \tag{7}$$

where r_t^{lv} and r_t^{av} are linear and angular velocity tracking scores (kernel width σ_{trk} in Table II), r_t^{al} is the alive bonus, r_t^{im} is imitation, and r_t^{τ}, $r_t^{\dot{a}}$, r_t^{ss} are nonnegative penalty magnitudes for torque, action rate, and stand-still behavior. The weights satisfy $w_{\text{lv}}, w_{\text{av}}, w_{\text{al}}, w_{\text{im}} > 0$ and $w_\tau, w_{\dot{a}}, w_{\text{ss}} < 0$, so the last three terms subtract cost. In short, r_t^{loc} defines *can the robot walk stably and follow commands?*

The proposed part is r_t^{style}, which defines *how the robot looks while walking*. At each episode, one label is sampled from (*Neutral*, *Happy*, *Sad*); we write $\mathbb{I}_{\text{N}}, \mathbb{I}_{\text{H}}, \mathbb{I}_{\text{S}} \in \{0, 1\}$ for the corresponding indicators (exactly one equals one). The command gate g_t satisfies $g_t \approx 0$ under very small locomotion commands so that style rewards do not interfere with balance at standstill.

Let q_t^{hp} and $\dot{q}_t^{\text{hp}}$ denote head-pitch angle and angular velocity (superscript $^{\text{hp}}$ is the joint tag). Define the head-pose closeness score

$$\phi(q; \mu, \sigma_{\text{hp}}) = \exp\left(-\frac{(q-\mu)^2}{\sigma_{\text{hp}}^2}\right), \tag{8}$$

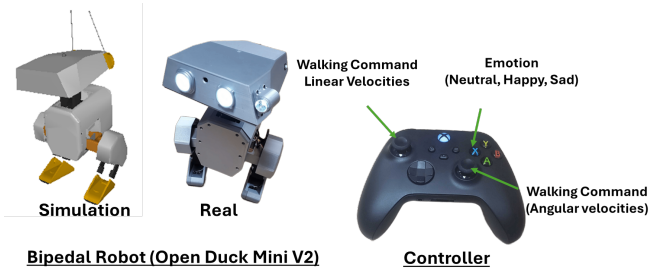

Fig. 3. Open Duck Mini V2 and Controller

where $\phi(\cdot)$ is bounded and smooth in q, μ is the target angle, and σ_{hp} is the tolerance width. The activity score is

$$\eta(\dot{q};\beta) = \tanh\left(\frac{|\dot{q}|}{\beta}\right), \tag{9}$$

where $\eta(\cdot)$ is bounded in $\dot{q}$ and β is the activity scale. The weighted style reward uses the coefficients w_{hn}–w_{ha} in Table II:

$$\begin{aligned} r_t^{\text{style}} = g_t\Big(&\mathbb{I}_{\text{N}}\, w_{\text{hn}}\, \phi(q_t^{\text{hp}};\mu_{\text{N}},\sigma_{\text{hp}}) \\ &+ \mathbb{I}_{\text{H}}\left(w_{\text{hu}}\, \phi(q_t^{\text{hp}};\mu_{\text{H}},\sigma_{\text{hp}}) + w_{\text{ha}}\, \eta(\dot{q}_t^{\text{hp}};\beta)\right) \\ &+ \mathbb{I}_{\text{S}}\left(w_{\text{hd}}\, \phi(q_t^{\text{hp}};\mu_{\text{S}},\sigma_{\text{hp}}) - w_{\text{ha}}\, \eta(\dot{q}_t^{\text{hp}};\beta)\right)\Big). \end{aligned} \tag{10}$$

where $\mu_{\text{N}}, \mu_{\text{H}}, \mu_{\text{S}}$ are neutral/upward/downward head-pitch targets, and the gate threshold δ in Table II sets $g_t \approx 0$ when the locomotion command norm is below δ.

For numerical stability, angle-closeness terms are bounded and smooth around each target, and activity terms are also bounded. Therefore, the final design cleanly separates responsibilities: the OSS-based r_t^{loc} preserves locomotion competence, while the proposed r_t^{style} induces interpretable style differences (*neutral*, *upward/lively*, *downward/subdued*).

E. Domain Randomization and Training Protocol

We train with vectorized simulation under a unified style-conditioned setting. To reduce sim-to-real mismatch, we apply domain randomization at reset and during rollout: sensor noise (gyro, accelerometer, gravity, joint signals), action/IMU delay, randomized initial base pose, and random external pushes. The episode terminates on fall or invalid physics states, and command/style resampling is periodically refreshed to increase behavioral coverage.

The final objective is to maximize the expected discounted return

$$J(\theta) = \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t r_t\right]. \tag{11}$$

Here, $J(\theta)$ is the training objective, $\mathbb{E}[\cdot]$ denotes expectation over trajectories induced by π_θ, $\gamma \in (0,1]$ is the discount factor, and T is the episode horizon. After training, the policy is exported to ONNX and evaluated under the same observation layout used in training, ensuring reproducible and deployment-consistent inference.

TABLE I

CORE DIFFERENCE BETWEEN THE BASELINE AND EMOLO

Item	Baseline	EmoLo (Proposed)
Policy input	101 dims	104 dims
Style code	Not included	Included
Reward	r_t^{loc} only	$r_t^{\text{loc}} + r_t^{\text{style}}$

TABLE II

TRAINING PARAMETERS

Symbol	Description	Value
w_{lv}	Linear velocity tracking weight	2.5
w_{av}	Angular velocity tracking weight	2.0
w_{al}	Alive bonus weight	20.0
w_{im}	Imitation weight	1.0
w_τ	Torque penalty weight	-1.0×10^{-3}
$w_{\dot{a}}$	Action-rate penalty weight	-0.5
w_{ss}	Stand-still penalty weight	-0.2
w_{hn}	Style: neutral head	2.0
w_{hu}	Style: happy head-up	2.0
w_{hd}	Style: sad head-down	2.0
w_{ha}	Style: head activity	0.6
σ_{trk}	Velocity tracking kernel width	0.01
δ	Style gate threshold (command norm)	0.03
β	Head activity scale	0.5 rad/s
μ_{N}	Neutral head-pitch target	0.0 rad
μ_{H}	Happy head-pitch target	0.3 rad
μ_{S}	Sad head-pitch target	-0.3 rad
σ_{hp}	Head-pose kernel width	0.05

IV. EXPERIMENTS

A. Simulation Setup

We evaluate the proposed style-conditioned policy together with a baseline policy that does not use style conditioning. Table I summarizes the core differences between the locomotion-only baseline and the proposed style-conditioned method. The proposed method uses the observation and reward design, namely a 104-dimensional observation including the style one-hot code and the combined reward

$$r_t = r_t^{\text{loc}} + r_t^{\text{style}}. \tag{12}$$

In contrast, the baseline removes the style input and uses only the locomotion objective,

$$r_t^{\text{base}} = r_t^{\text{loc}}. \tag{13}$$

Accordingly, the baseline policy receives a 101-dimensional observation vector obtained by excluding the 3-dimensional style one-hot code from the policy input:

$$\begin{aligned} o_t^{\text{base}} = \big[&\omega_t^{\text{imu}},\ a_t^{\text{imu}},\ c_t,\ \Delta q_t,\ \dot{q}_t, \\ &a_{t-1}, a_{t-2}, a_{t-3}, u_{t-1}, \kappa_t, \phi_t\big]. \end{aligned} \tag{14}$$

Both policies output a 14-dimensional action vector and are trained under the same simulator, command ranges, control frequency, domain randomization settings, and termination conditions.

Reward-related parameters are listed in Table II, where the style-specific terms are used only in the proposed method. For robustness, training also uses delay randomization, sensor noise injection, random initial-state perturbation, and random external pushes, as described in Sec. III-E. The baseline comparison is conducted in simulation only, while

Fig. 4. Simulation comparison between the baseline and the proposed style-conditioned policy. Representative snapshots show that the baseline maintains locomotion without explicit style modulation, while the proposed policy produces distinct head-pitch postures for Happy, Neutral, and Sad.

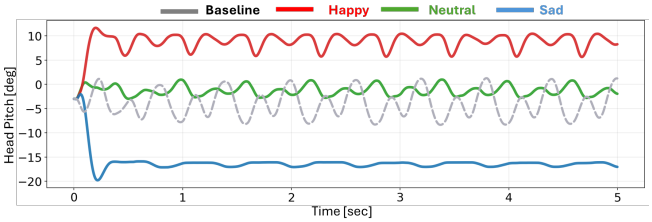

Fig. 5. Simulation head-pitch trajectories for the baseline and the three style conditions of the proposed method.

real-world experiments are reported only for the proposed method.

Simulation evaluation addresses three questions:

- **Robust locomotion:** Does the policy complete the trajectory without falling?
- **Style separability:** Do the three styles yield distinct head-pitch time-series signatures?
- **Baseline comparison:** Compared with the locomotion-only baseline, does the proposed method yield clearer style-dependent head-pitch modulation?

We record the head-pitch time series q_t^{hp} and log binary fall events and trial completion. The evaluation protocol is a fixed forward-walking task: each rollout commands forward motion for 5 s at constant positive longitudinal velocity, with zero lateral and yaw velocity commands. For the proposed method, we run three separate trials that differ only in the active style label (*Neutral*, *Happy*, *Sad*) so that style effects are isolated from command variation. For the baseline, we run the same task under the same command condition. A trial is counted as successful if the robot remains upright and walking for the full 5 s horizon without fall termination. During each rollout we inspect temporal variability of q_t^{hp}.

B. Simulation Results

Under the simulation setup (Sec. IV-A), both the baseline and the proposed method completed the full 5 s forward-walking evaluation without fall termination. This indicates that the additional style-conditioning mechanism does not compromise locomotion stability under the tested condition. Representative snapshots are shown in Fig. 4. The baseline exhibits a single locomotion pattern without explicit style-dependent modulation, whereas the proposed method produces visually distinct head-pitch postures for *Happy*, *Neutral*, and *Sad* during the same forward-walking task.

These differences are further clarified by the head-pitch trajectories q_t^{hp} in Fig. 5. The baseline remains within a limited range around its nominal behavior and does not exhibit the clear style-specific separation observed in the proposed method. In contrast, the proposed policy yields three distinct regimes. For *Happy*, the trajectory rapidly transitions to a clearly positive head-pitch region and remains elevated throughout the walking motion. In addition to this upward bias, the signal exhibits pronounced periodic oscillations, indicating active head motion superimposed on the gait cycle. This behavior is consistent with the intended combination of an upward pose target and an activity-promoting reward term.

For *Neutral*, the trajectory remains centered near the nominal range around zero, with moderate oscillatory variation over time. Compared with *Happy*, the mean head-pitch level is substantially lower and the oscillations are less pronounced, producing a visually calmer head motion during walking. At the same time, the signal is not completely static; instead, it follows the walking rhythm with modest periodic fluctuations, which is consistent with a neutral reference behavior rather than explicit suppression of motion.

For *Sad*, the trajectory shows the opposite tendency. Af-

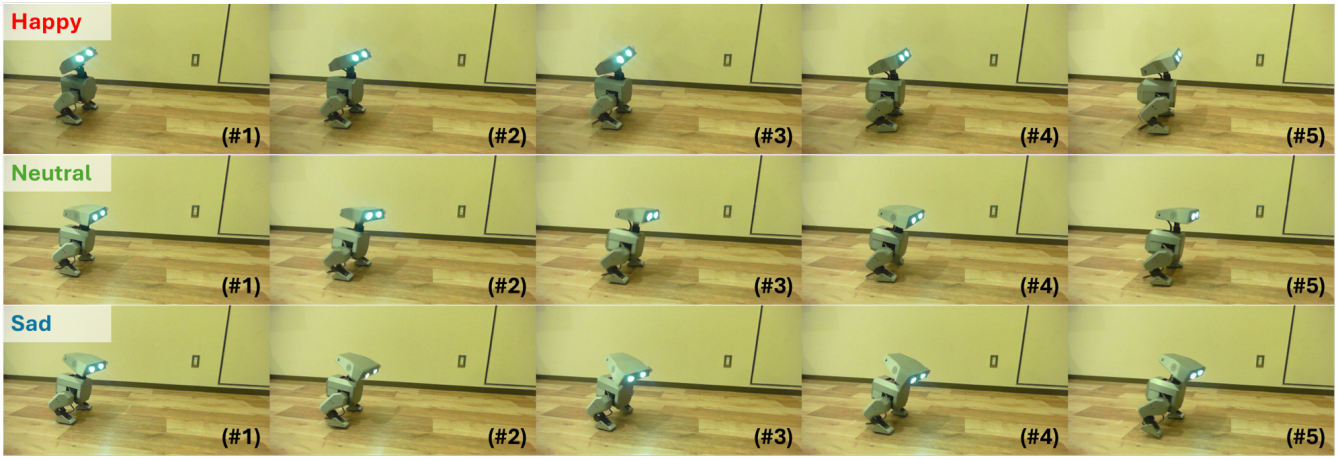

Fig. 6. Representative hardware snapshots under the three style conditions during the shared forward-walking task. Across the same walking sequence, *Happy* maintains an upward head posture, *Neutral* remains near the nominal posture, and *Sad* shows a downward head posture.

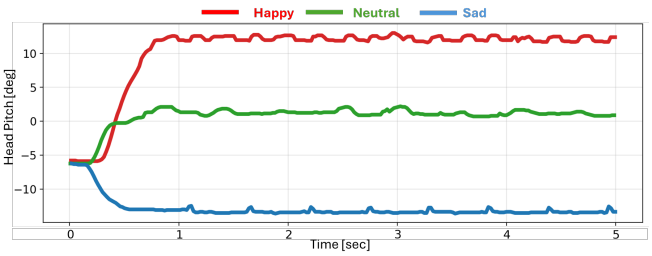

Fig. 7. Real World Results of Head Pitch

ter a brief transient, the head pitch quickly moves into a clearly negative region and then remains low for the rest of the rollout. The temporal variation is also smaller than in *Happy*, yielding a more subdued motion pattern overall. This downward and less active behavior matches the intended effect of the downward pose target together with the activity-suppressing component of the style reward.

Taken together, the simulation results show three important properties. First, the baseline confirms that stable forward locomotion can be achieved without style conditioning. Second, the proposed controller preserves this locomotion competence across all tested styles, indicating that the added style objective does not destabilize the underlying locomotion behavior in this task. Third, explicit style input and style reward are necessary to generate clearly separated head-pitch regimes with distinct motion characteristics. In this sense, the simulation results support the claim that the proposed design induces interpretable, emotion-inspired style variation in head-related walking behavior while maintaining successful locomotion under the tested conditions.

C. Real Experiment Setup

The real experiment uses the same trained policy exported to ONNX as in simulation. Deployment follows a sim-to-real protocol without on-hardware fine-tuning or retraining, in order to test whether the learned controller transfers directly to the physical Open Duck Mini V2 platform (Fig. 3). The real-world experiments are conducted only for the proposed method.

The command task mirrors the simulation protocol: each trial consists of 5 s of constant forward walking, repeated for each style label while all other conditions are held fixed. We stream the same observation layout used during training and execute ONNX inference on board in closed loop.

D. Real Experiment Results

The exported ONNX policy ran reliably on hardware and generated continuous control commands throughout the evaluation window. Representative snapshots are shown in Fig. 6. As in simulation, the visual appearance of the head posture differs systematically across styles: *Happy* maintains a clearly upward-oriented head posture, *Neutral* remains near an intermediate posture, and *Sad* exhibits a consistently downward-oriented head posture over the walking sequence.

The logged head-pitch trajectories in Fig. 7 show that these qualitative differences are also reflected in the measured time series. After a short initial transition, *Happy* rises from the initial posture into the highest positive head-pitch region and then remains elevated for the rest of the trial. *Neutral* converges to an intermediate range near the nominal posture and stays clearly below *Happy*. In contrast, *Sad* remains in the lowest region throughout the rollout, preserving a distinctly downward head configuration. Thus, the relative ordering among the three styles is maintained on hardware in the same direction as observed in simulation.

Taken together, the real experiment supports two conclusions. First, the trained controller is directly deployable on the physical platform via ONNX without additional tuning. Second, the intended emotion-inspired style differences remain observable on hardware while preserving successful forward locomotion under the shared walking protocol. These results suggest that lightweight, reward-level style conditioning can provide interpretable expressive modulation on a low-cost biped without sacrificing practical deployability.

V. CONCLUSION

This paper presented a style-conditioned reinforcement learning approach for Open Duck Mini V2 that preserves a strong locomotion backbone while enabling emotion-inspired walking through a discrete style label (*Neutral*, *Happy*, *Sad*).

Simulation results showed stable locomotion across all styles, with head-pitch trajectories reflecting the intended ordering (*Happy* more upright and active, *Sad* lower and calmer, *Neutral* in between). Hardware experiments reproduced the same ordering and demonstrated direct sim-to-real transfer via ONNX without additional tuning. A simulation-only baseline further confirmed that while stable walking can be achieved without style conditioning, explicit style input and reward are required to generate clear, interpretable expressive modulation. These results support the central claim that expressive behavior can be integrated into learned legged control through a minimal and interpretable reward design, without restructuring the locomotion pipeline.

Our work has limitations. Style expression is currently localized to head-pitch, and full-body stylistic variation remains unexplored. Future work includes extending style dimensions (e.g., tempo or upper-body coordination), introducing scaling to more complex real-world tasks. This line of work aims to make expressive legged behavior practical on resource-constrained robotic platforms.

REFERENCES

[1] M. Fujita and K. Kageyama, "An open architecture for robot entertainment," in *Proceedings of the First International Conference on Autonomous Agents*, ser. AGENTS '97. New York, NY, USA: Association for Computing Machinery, 1997, p. 435–442. [Online]. Available: https://doi.org/10.1145/267658.267764

[2] Q. Liao, B. Zhang, X. Huang, X. Huang, Z. Li, and K. Sreenath, "Berkeley humanoid: A research platform for learning-based control," 2024. [Online]. Available: https://arxiv.org/abs/2407.21781

[3] G. Venture and D. Kulić, "Robot expressive motions: A survey of generation and evaluation methods," *J. Hum.-Robot Interact.*, vol. 8, no. 4, Nov. 2019. [Online]. Available: https://doi.org/10.1145/3344286

[4] V. Narayanan, B. M. Manoghar, R. P. RV, and A. Bera, "Ewarenet: Emotion-aware pedestrian intent prediction and adaptive spatial profile fusion for social robot navigation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 7569–7575.

[5] W. Zhu, A. Raju, A. Shamsah, A. Wu, S. Hutchinson, and Y. Zhao, "Emobipednav: Emotion-aware social navigation for bipedal robots with deep reinforcement learning," *IEEE/ASME Transactions on Mechatronics*, pp. 1–13, 2026.

[6] H. Liu, M. Zhu, A. M. F. Alvarez, Y. H. Lo, C. Ku, F. Parres, J. Quan, C. Togashi, A. Navghare, Q. Wang, and D. W. Hong, "From screen to stage: Kid cosmo, a life-like, torque-controlled humanoid for entertainment robotics," in *2025 IEEE-RAS 24th International Conference on Humanoid Robots (Humanoids)*, 2025, pp. 677–684.

[7] R. Grandia, E. Knoop, M. Hopkins, G. Wiedebach, J. Bishop, S. Pickles, D. Müller, and M. Bächer, "Design and control of a bipedal robotic character," in *Robotics: Science and Systems XX*, ser. RSS2024. Robotics: Science and Systems Foundation, Jul. 2024. [Online]. Available: http://dx.doi.org/10.15607/RSS.2024.XX.103

[8] D. Müller, E. Knoop, D. Mylonopoulos, A. Serifi, M. A. Hopkins, R. Grandia, and M. Bächer, "Olaf: Bringing an animated character to life in the physical world," 2025. [Online]. Available: https://arxiv.org/abs/2512.16705

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

[10] 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, Jan. 2019. [Online]. Available: http://dx.doi.org/10.1126/scirobotics.aau5872

[11] 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. [Online]. Available: https://www.science.org/doi/abs/10.1126/scirobotics.abk2822

[12] X. B. Peng, E. Coumans, T. Zhang, T.-W. Lee, J. Tan, and S. Levine, "Learning agile robotic locomotion skills by imitating animals," 2020. [Online]. Available: https://arxiv.org/abs/2004.00784

[13] A. Pirrone, "Open Duck Mini: A miniature version of the BDX droid," 2025. [Online]. Available: https://github.com/apirrone/Open_Duck_Mini

[14] F. Chen, R. Wan, P. Liu, N. Zheng, B. Wang, and B. Zhou, "Vmts: Vision-assisted teacher-student reinforcement learning for multi-terrain locomotion in bipedal robots," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025, pp. 4759–4766.

[15] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 5026–5033.

[16] D. Kim, H. Lee, J. Cha, and J. Park, "Bridging the reality gap: Analyzing sim-to-real transfer techniques for reinforcement learning in humanoid bipedal locomotion," *IEEE Robotics & Automation Magazine*, vol. 32, no. 1, pp. 49–58, 2025.

[17] V. Tsounis, G. Maloisel, C. Schumacher, R. Grandia, A. Serifi, D. Müller, C. Amevor, T. Widmer, and M. Bächer, "Kamino: Gpu-based massively parallel simulation of multi-body systems with challenging topologies," 2026. [Online]. Available: https://arxiv.org/abs/2603.16536

[18] G. Mothish, K. Rajgopal, R. Kola, M. Tayal, and S. Kolathaya, "Stoch biro: Design and control of a low cost bipedal robot," 2023. [Online]. Available: https://arxiv.org/abs/2312.06512

[19] Y. Huang, Y. Zeng, and X. Xiong, "Stride: An open-source, low-cost, and versatile bipedal robot platform for research and education," 2024. [Online]. Available: https://arxiv.org/abs/2407.02648

[20] Y. Chi, Q. Liao, J. Long, X. Huang, S. Shao, B. Nikolic, Z. Li, and K. Sreenath, "Demonstrating berkeley humanoid lite: An open-source, accessible, and customizable 3d-printed humanoid robot," 2025. [Online]. Available: https://arxiv.org/abs/2504.17249

[21] B. Xia, B. Li, J. Lee, M. Scutari, and B. Chen, "The duke humanoid: Design and control for energy efficient bipedal locomotion using passive dynamics," 2025. [Online]. Available: https://arxiv.org/abs/2409.19795

[22] P. Allgeuer, H. Farazi, M. Schreiber, and S. Behnke, "Child-sized 3d printed igus humanoid open platform," in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, 2015, pp. 33–40.

[23] H. Shi, W. Wang, S. Song, and C. K. Liu, "Toddlerbot: Open-source ml-compatible humanoid platform for loco-manipulation," 2025. [Online]. Available: https://arxiv.org/abs/2502.00893

[24] M. Fujita, "Aibo: Towards the era of digital creatures," in *Robotics Research*, J. M. Hollerbach and D. E. Koditschek, Eds. London: Springer London, 2000, pp. 315–320.

[25] L. Roy, E. A. Croft, A. Ramirez, and D. Kulić, "Llm-driven expressive robot motion via proxy-based optimization," *IEEE Robotics and Automation Letters*, vol. 11, no. 4, pp. 4561–4568, 2026.

[26] X. B. Peng, P. Abbeel, S. Levine, and M. van de Panne, "Deepmimic: Example-guided deep reinforcement learning of physics-based character skills," *ACM Trans. Graph.*, vol. 37, no. 4, pp. 143:1–143:14, Jul. 2018. [Online]. Available: http://doi.acm.org/10.1145/3197517.3201311

[27] M. Heyrman, C. Li, V. Klemm, D. Kang, S. Coros, and M. Hutter, "Multi-domain motion embedding: Expressive real-time mimicry for legged robots," 2025. [Online]. Available: https://arxiv.org/abs/2512.07673

[28] Z. Li, C. Cummings, and K. Sreenath, "Animated cassie: A dynamic relatable robotic character," 2020. [Online]. Available: https://arxiv.org/abs/2009.02846