ViBe: Visual Behavior Adaptation for Perceptive Humanoid Whole-Body Control

Lokesh Krishna, Sarvesh Venkatesan, An Zhang, Quan Nguyen
University of Southern California
project page: lok-i.github.io/vibe-control

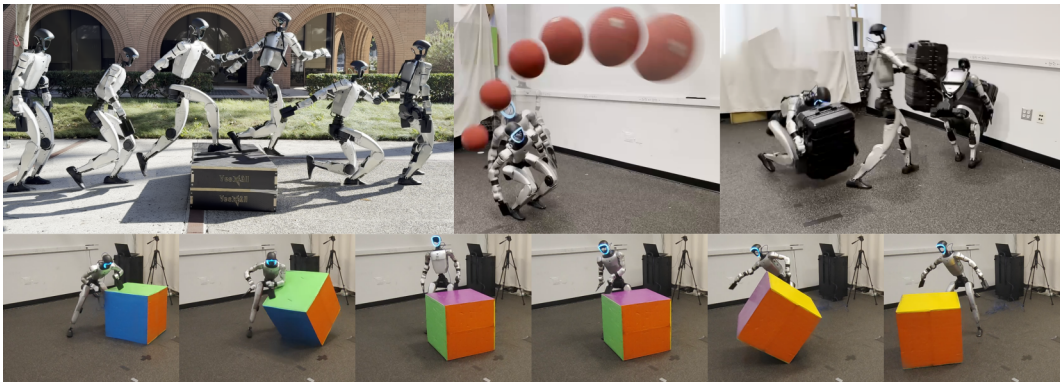


Figure 1: **Visual behavior adaptation.** A single post-training recipe adapts a pre-trained humanoid tracker to perceptive parkour, dodgeball, loco-manipulation and cube reorientation.

Abstract: Motion tracking provides a scalable recipe for humanoid whole-body control. By design, the resulting trackers lack exteroceptive feedback hence reacting to the environment remains the responsibility of a higher-level planner. Existing perceptive controllers train geometry-only encoders from scratch, trading semantics for sim-to-real ease, and typically rely on teacher-student distillation for a task of interest. We present ViBe, a post-training framework for adapting motion trackers to perceptive control tasks. We leverage pre-trained visual encoders with a multi-query extractor module to learn task-relevant perceptive feedback. This feedback is grafted onto the tracker's input via low-rank adapters, enabling parameter-efficient fine-tuning. Given a task reward and a reference dataset, this modular controller can be adapted directly via policy optimization. Across four tasks, ViBe shows zero-shot sim-to-real transfer spanning perceptive walking on curbs and parkour, Repose Cube, omni-object loco-manipulation, and dodgeball, with visually robust performance across outdoor, low-light, and RGB distractor conditions. Finally, we solve a goal-oriented Repose Cube task with a deliberately simple planner, demonstrating the efficacy of perceptive controllers, adapted by our approach.

Keywords: Humanoid whole-body control, perceptive control, policy adaptation

1 Introduction

Humanoid whole-body control has found an effective and scalable approach: learn to mimic large collections of human motion on flat terrain. Increasing the diversity of reference motions, model capacity, and training compute produces controllers that preserve natural motion across a broad behavioral repertoire [1, 2]. These trackers are useful motor priors, but they are intentionally designed blind to their environment. When a tracker encounters an environment constraint or external dynamics, the tracker has no observation to react or adapt its behavior to solve the task.

A conventional hierarchy resolves this mismatch above the controller. Perception builds an environment representation, a planner modifies the reference, and the tracker executes it. This separation has enabled perceptive locomotion [3] and interactive whole-body control [1], yet it places every environment-dependent correction at the planning level. Such an architecture limits the lower-level modules from controlling the contact placement to reactively interact with the environment, expecting a plan to be sufficiently accurate. Such an architecture results in missing local corrections and visual reflexes like correcting foot placement over curbs, adjusting a grasp on an object, or dodging an incoming ball.

Learning-based perceptive control exposes these corrections directly to a controller. Recent systems traverse terrain [4, 5, 6] and interact with non-flat scenes [7], and learn visual loco-manipulation [8]. Most approaches train task-specific perceptive encoders and trackers and rely on teacher-student distillation [9, 10, 4, 6]. On the other hand, attention-based encoders have recently shown that robot state can select control-relevant geometric feedback from dense observations for locomotion [11, 12]. However, a generality-preserving approach that combines a pre-trained vision encoder with a pre-trained whole-body tracker, adapting them across diverse humanoid perceptive-control tasks remains largely unexplored.

Our approach introduces an interface between the two pre-trained components. A frozen vision encoder produces dense tokens, from which a compact cross-attention extractor selects task-relevant visual feedback using the task command, global CLS token, and robot proprioception. Low-rank adapters inject this feedback into the frozen whole-body tracker, forming a visual bypass that leaves both pre-trained components intact. Given a reference-motion dataset and a task reward, we train the extractor and adapters through policy optimization, without teacher-student distillation or auxiliary representation losses.

Our contributions are twofold:

- a visual behavior adaptation framework that bridges a frozen vision encoder and a frozen whole-body tracker through a task-conditioned extractor and low-rank adapters, post-trained directly with reinforcement learning.
- zero-shot sim-to-real transfer across four tasks: perceptive locomotion, object reorientation, loco-manipulation, and dodgeball, with controlled ablations

2 Related Works

Humanoid whole-body control. Reference tracking turns motion data into whole-body humanoid trackers [13, 2]. Scaling motion data and model capacity yields a broad whole-body prior [1, 14]. Recent extensions add history-based adaptation for tracking under dynamic disturbances [15] or model environment-conditioned behavior for interaction-aware control [16]. Our work retains the pre-trained tracker and learns a perceptive adaptation.

Perceptive robot control. Classical systems separate perception, planning, and control through strict hierarchies and explicit contracts [3]. Learning-based approaches need not follow these architectural constraints and can learn pathways that bypass such interfaces. Existing approaches map exteroception to actions for quadruped locomotion [17, 18], humanoid terrain traversal [4, 5, 6, 19], dexterous manipulation [20], and whole-body interaction [7, 8]. A common sim-to-real recipe trains an expert with privileged observations and distills it into a student with a perceptive encoder trained from scratch. ViBe instead post-trains a deployable visual actor directly via policy optimization.

Visual representations and attention. Pre-trained encoders provide transferable visual features without learning perception from each task's simulated images [21, 22, 23]. For control, attention can use robot state to select relevant regions of a terrain map or dense depth observation [11, 12]. ViBe applies the same selection principle to visual tokens shared across perceptive whole-body

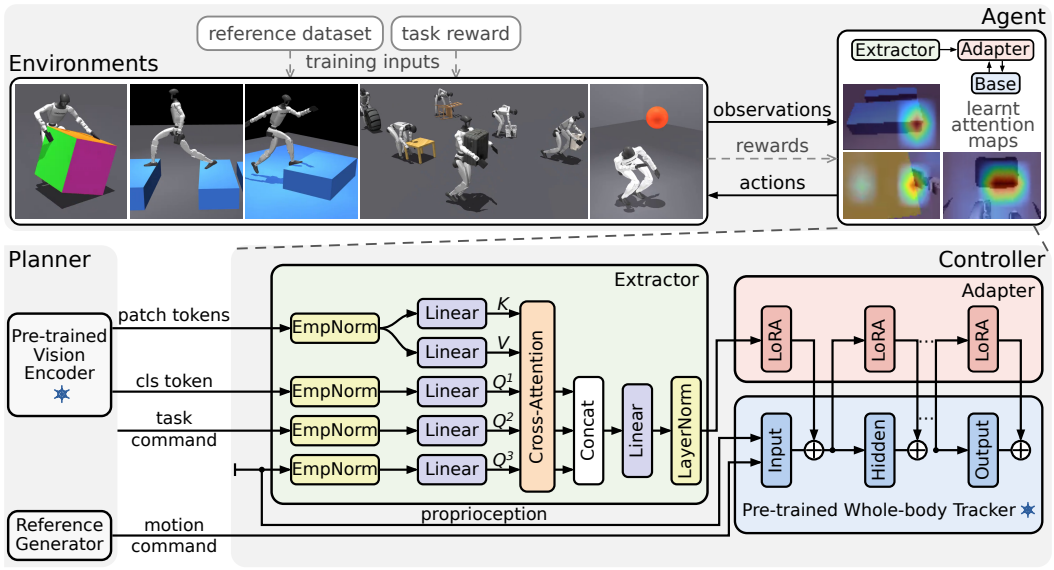


Figure 2: **ViBe bridges visual and motor priors.** *Top:* Each task trains the same agent architecture from a reference-motion dataset and task reward. *Bottom:* An extractor uses global image context, task commands, and proprioception to attend to patch tokens from a frozen vision encoder. The percept is grafted via LoRA adapters to the frozen whole-body tracker.

control tasks. The extractor neither predicts privileged observations nor minimizes auxiliary learning objectives but is trained only through the downstream task objective.

3 Approach

As a post-training framework, ViBe learns only the compact extractor–adapter pathway, leaving both pre-trained priors intact as shown in Fig. 2.

3.1 Formulation

We formulate each downstream task as an episodic Markov decision process with partial actor observations. The deployable actor observation contains the robot's proprioceptive history, a future window of the reference motion, the egocentric camera image, and an optional task command. The policy outputs actions to construct PD targets for each actuated degree of freedom [1]. Following [13], the reward combines motion tracking with a task reward like object tracking, object goal, or avoidance:

$$\max_{\theta} \quad \mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{T} \gamma^t \left(r_{\text{track}} + r_{\text{task}}\right)\right]. \tag{1}$$

We use an asymmetric actor–critic [24]: privileged observations are available to the critic, while the actor remains consistent with deployment. We optimize the three-module agent with PPO [25, 26]. For the base whole-body tracker, we use SONIC [1] and adapt its decoder. We zero-initialize the adapters so the policy initially matches the base tracker, and keep the policy's action standard deviation fixed throughout adaptation. ViBe preserves the base tracker input-outputs and adds three components: a frozen visual encoder that maps the egocentric image to visual tokens, an extractor that selects task-relevant visual feedback, and low-rank adapters that inject this feedback into the tracker. We train one adapted policy per task using the same architecture and training recipe.

Table 1: **Overview of Task Configurations.**

Task	**Reference, Num. Clips**	**Task Reward**	**Task Command**
Repose Cube	Custom in-house, 78	Up-face (ours)	Face one-hot
Walk / Parkour	GRAIL [27], 63 OmniRetarget [28], 9	Tracking [13]	Root twist
Omni-object loco-manipulation	Custom in-house, 6	Object / contact [29]	Goal pose
Dodgeball	Nominal pose, 1	Clearance [30]	—

3.2 Task-based Visual Extractor

The extractor was designed to address the question: which image patches matter to the downstream task? Our default vision encoder is a Theia-Tiny ViT [21], operating on RGB images, returning a spatial grid of patch tokens and one global token. We use the patch tokens as keys and values for a single-head cross-attention layer [31]. Separate, normalized query projections are formed from three signals: the global image token, robot proprioception and optionally the task command.

Each query produces an attention-weighted summary of the patch grid, as shown in Fig. 2. We concatenate these summaries, project them to a fixed 128-dimensional *percept*, and apply layer normalization. This fixed output size decouples the controller interface from image resolution and the encoder's latent dimensions. Note that the query signals guide patch selection but do not enter the percept directly. While proprioception and reference commands retain their direct paths to the controller, the task commands enter only through visual selection.

The *percept* conditions rank-16 LoRA adapters in the frozen tracker's decoder [32]. We apply PPO updates only to the extractor and adapters, keeping all other modules frozen and restricting adaptation to task-specific corrections of the motion prior.

3.3 Reference-Phase Annealing

We observed that policies trained with uniform Reference State Initialization (RSI) can specialize to short motion segments but fail to traverse the entire reference. Additionally, the reference phase often tracks task progress, with later phases corresponding to later stages of task completion [33]. These observations motivate reference-phase annealing as a complementary curriculum. Let $\phi^{\max}$ denote the upper support of the normalized-phase distribution. At policy iteration k, given an annealing period of K, we set $\phi_k^{\max}$ and sample the RSI phase ϕ_0 as

$$\phi_k^{\max} = \max\left(0, 1 - \frac{k}{K}\right), \qquad \phi_0 \sim \mathrm{Uniform}(0, \phi_k^{\max}). \tag{2}$$

At the start of training, resets span the full reference, making the curriculum identical to uniform RSI. As training proceeds, the interval contracts toward frame zero. After K updates, every episode begins at the start and thus forces mastering the full reference.

4 Results

4.1 Performance and Comparisons

Experimental setup. We evaluate on four perceptive control tasks on a Unitree G1. Table 1 summarizes the key defining factors: reference dataset, task-specific reward, and task command. Together, the tasks expose complementary forms of visual adaptation:

1. **Repose Cube.** Inspired by vision-based in-hand reorientation [34], we define a whole-body variant: the humanoid tracks the reference while ensuring the cube tracks the object reference to result in the end-frame's face is on top.

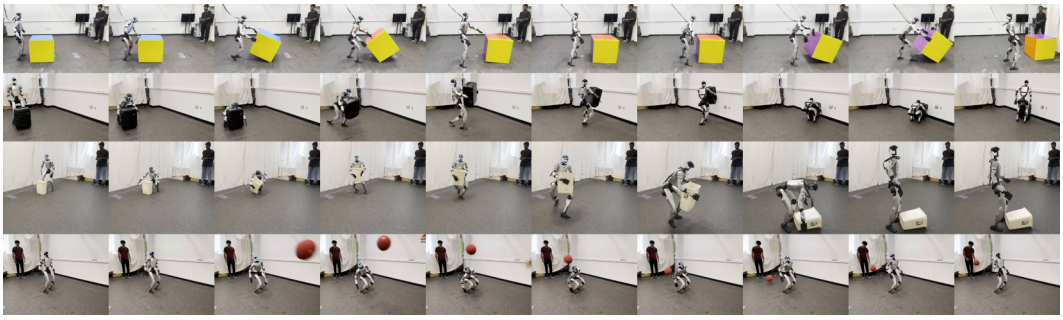


Figure 3: **ViBe adapts reference motion to task dynamics.** Zero-shot hardware rollouts show cube reorientation, suitcase transport, trash-can transport, and dodgeball (top to bottom). Each row contains ten equally spaced frames from one uninterrupted fixed-camera rollout.

2. **Perceptive Walk and Parkour.** The two variants provide complementary motion-terrain pairings. GRAIL maps many walking clips per terrain, while each OmniRetarget parkour clip is paired with its own terrain and each task environment has 9 distinct terrains.
3. **Omni-object Loco-manipulation.** We place distinct objects in one environment and train a shared policy across them. This was to discourage memorization of a single geometry and makes the extractor attend to the geometry required for each interaction.
4. **Dodgeball.** Finally, dodgeball separates task success from reference tracking. The reference remains a nominal standing pose, but avoiding a moving ball requires dynamic perceptive adaptation away from that pose.

We implement each task in mjlab [35], built for MuJoCo Warp [36, 37]. Thanks to its GPU batch renderer [38], training at the necessary resolution (112×63) ends within a reasonable wall time (under two days) across different GPUs: NVIDIA RTX 3090, RTX 5090, and L40S. Policies run at a control rate of 50 Hz, which is also the frame rate of the head-mounted Intel RealSense D435i camera at the same resolution. Unless stated otherwise, every task uses exactly the same architecture: a frozen Theia-Tiny encoder, the learnable extractor and rank-16 adapters. We randomize camera extrinsics, lighting, ground appearance, robot parameters, and external perturbations necessary for sim-to-real transfer.

Zero-shot transfer. Fig. 1 summarizes the hardware evaluation. Post adaptation, ViBe preserves the essence of the reference motion while learning minimal desirable deviation in contact placement and motion to solve the task. Across tasks, spanning distinct control regimes: the robot adjusts foot placement on raised terrain and gains momentum for a jump, learns to reposition itself while carrying large objects, turns a cube through repeated contacts, and departs from a stationary reference during dodgeball (Fig. 3), separate task-specific policies trained with the same adaptation recipe.

Privileged observation vs. visual learning. We compare direct visual learning with privileged-observation training to validate whether post-training requires teacher-student distillation. Trained directly from visual feedback, ViBe reaches 89.6%, 95.5%, 90.3%, and 94.7% of the corresponding privileged policy on perceptive walk, perceptive parkour, omni-object loco-manipulation, and dodgeball, respectively. Across tasks, ViBe asymptotes privileged-observation performance without teacher action targets, indicating that teacher-student distillation is unnecessary for post-training in our setting.

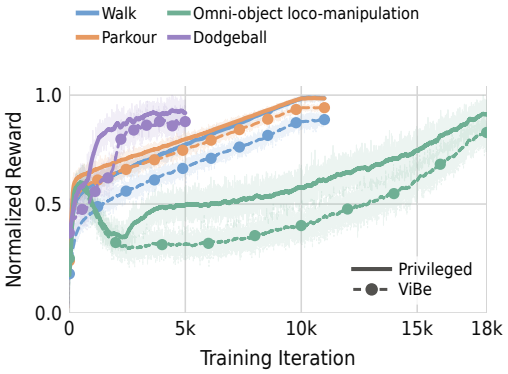
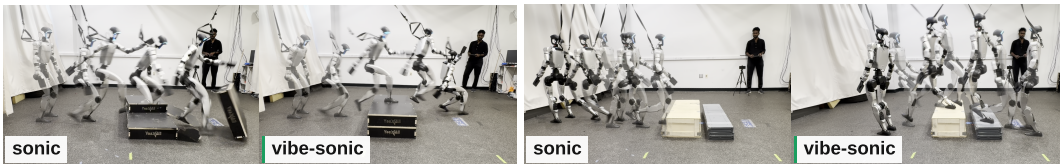


Figure 4: ViBe approaches privileged-observation performance.



(a) Perceptive Parkour (b) Perceptive Walk

Figure 5: **Visual adaptation enables terrain-aware tracking.** With the same SONIC base and reference clips, both in (a) parkour and (b) walk: the blind tracker (left) fails to traverse the terrain, whereas ViBe-SONIC (right) adjusts its motion and succeds in both behaviors.

Blind tracking vs. visual adaptation. The controlled comparison in Fig. 5 isolates the visual bypass. Both policies share the same base and reference motion, but only ViBe-SONIC observes the scene. On Perceptive Walk, the blind tracker collides with the curb, preventing ascent and causing global root drift; ViBe-SONIC instead adjusts its foot placement and traverses the curb. On Perceptive Parkour, ViBe-SONIC also tracks the motion velocity more closely, building the momentum needed to complete the jump. With the base and reference fixed, these differences reflect visual adaptation rather than tracker quality.

Visual robustness. Policies trained with the same frozen vision backbone remain effective under out-of-distribution visual conditions unseen during training. Fig. 6 shows perceptive walk outdoors in bright midday sunlight and repose cube with low light and RGB distractors. These trials demonstrate tolerance to static appearance shifts; but do not establish invariance to unmodeled dynamic physical distractors, which remain a limitation.

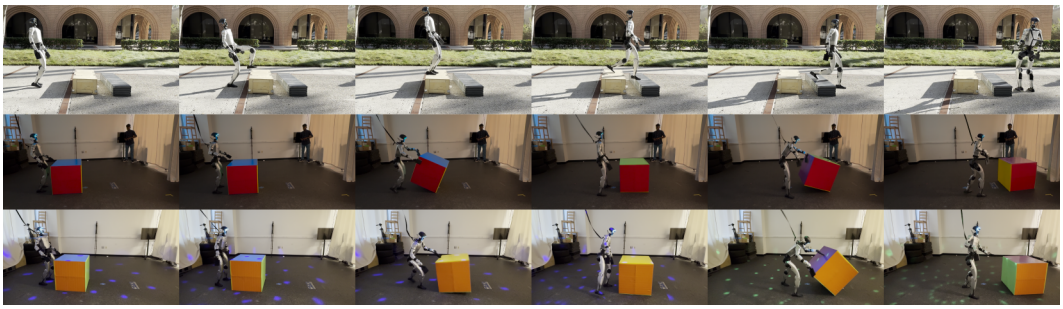


Figure 6: **Adapted policies tolerate substantial appearance shifts.** Zero-shot hardware rollouts show Perceptive Walk outdoors in direct sunlight(top), Repose Cube in low light (center), and Repose Cube under moving colored illumination (bottom)

4.2 Solving Tasks with a Planner

To demonstrate the efficacy of our perceptive control, we design a deliberately simple planner to solve the Repose Cube task. Like Sledgehammer and Sune in Rubik's Cube solving [39, 40], the planner repeatedly executes a fixed routine: right flip, right flip, front flip, until the requested color is on top. The planner has no dynamics model orforward predictive rollouts and merely chooses the clip from the fixed dataset. Since the adapted controller handles object localization, contact placement, and recovery from the stance mismatches through learnt visual cues, even a simple planner can solve the task. The hardware rollouts in Fig. 7 show three goal colors with more in the supplementary video. Thus, we demonstrate that a perceptive controller capable of local corrections can effectively solve tasks in conjunction with a higher-level planner.

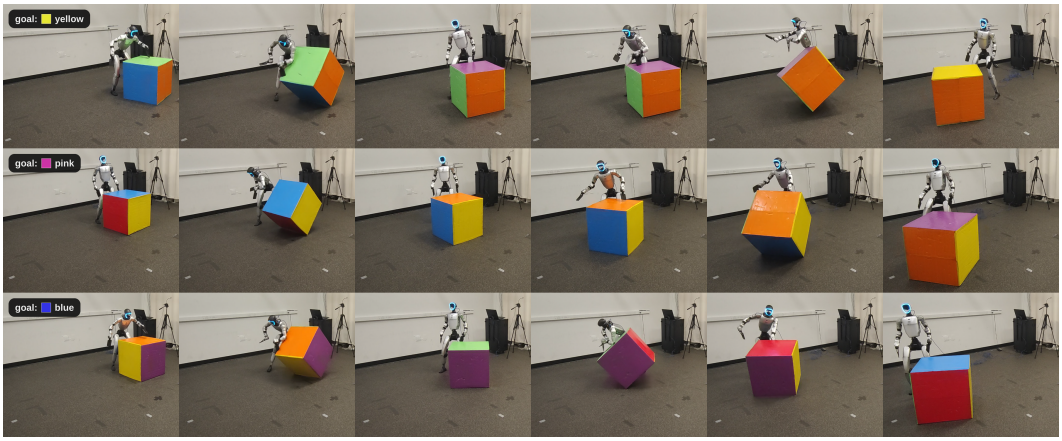


Figure 7: **A simple planner solves repose task** For yellow, pink, and blue goals (top to bottom), the planner selects fixed routine of reference clips while ViBe-adapted perceptive policies handle the rest. Each row shows one uninterrupted hardware rollout.

4.3 Ablations

We use the Repose task for controlled design ablations due to its well-defined success criterion: the requested face must lie within said threshold ($17.2°$) of vertical.

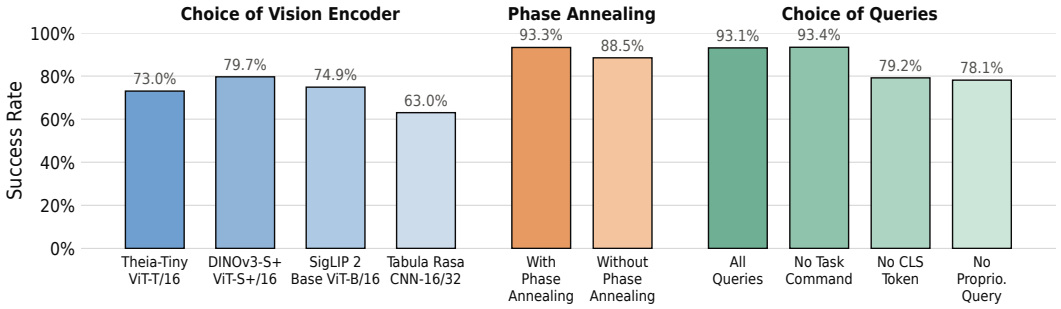


Figure 8: **Controlled ablations on Repose Cube.** Success rates vs choice of vision encoders (left), with and without phase annealing (center) and extractor queries (right)

Choice of vision encoder. To validate the flexibility of our approach, we ablate the choice of vision encoder. We compare the default Theia-Tiny [21] with DINOv3-S+ [22], SigLIP2-B [23], and a CNN trained from scratch [41]. After an equal number of policy updates, DINOv3-S+ achieves the highest success rate, followed by SigLIP2-B, Theia-Tiny, and the tabula-rasa CNN as shown in Fig. 8 (left). Success across the pre-trained backbones is tightly clustered, with a standard deviation of only 2.8%, while each outperforms the CNN trained from scratch. Thus, the extractor learns effective visual feedback across backbone choices.

Choice of queries. To ablate the contribution of each query, we remove one query at a time and compare the resulting success rates. With all three query rows, the policy achieves a success rate of 93.1% at $50k$ updates. Removing the CLS token or robot proprioception significantly reduces success by 15%, showing the significance of global context and robot's state in extracting task-relevant visual feedback. Interestingly, removing the task command query marginally increases success to 93.4%, which may be due to the nature of the chosen task-command for Repose task, demanding further investigation in the future across other tasks. Thus, the extractor benefits from the multi-query design.

Effect of phase annealing. Training with phase annealing marginally improves success rate from 88.5% to 93.3% as shown in Fig. 8 (right). However, qualitatively, we found the curriculum helps keep the learning dynamics stable across different tasks with varying dynamics and reward landscapes.

5 Conclusion and Limitations

We presented ViBe, a post-training framework for visual behavior adaptation of humanoid whole-body trackers by learning to extract task-relevant perceptive feedback from pre-trained vision encoders. Equipping a "blind" tracker with attention-based perception through low-rank adapters enables parameter-efficient RL fine-tuning, yielding visual policies that transfer zero-shot from simulation to hardware. We validate the same recipe across four tasks spanning non-flat terrain traversal, omni-object loco-manipulation, dodgeball, and Repose Cube. Together, these experiments demonstrate that perceptive whole-body control can be achieved by learning only the intermediate connection from scratch while retaining the pre-trained modules, providing a general recipe for perceptive humanoid control.

In its current form, we note two limitations of our work. First, while the resulting policies are visually robust across a wide range of lighting conditions and surface appearances, dynamic distractors can still trigger false responses. In dodgeball, for example, the policy sometimes mistakes the thrower's head for the ball and dodges unnecessarily. Future work will target learning necessary invariances through additional domain-randomization events. Second, the planner and controller are not trained in a closed loop. The planner may therefore select a reference whose transitions fall outside the adapter's training distribution. Although retaining the frozen tracker preserves reasonable tracking, task-optimal performance would benefit from training with a closed-loop reference generator [42, 43].

Acknowledgments

We thank Junchao Ma for helping with the hardware enhancements and Shravan Shenoy for early contributions to this work.

References

[1] Z. Luo, Y. Yuan, T. Wang, C. Li, F. Castañeda, S. Chen, Z.-A. Cao, J. Li, D. Minor, Q. Ben, J. Park, D. Sami, Z. Wang, X. Da, R. Ding, C. Hogg, L. Song, E. Lim, E. Jeong, T. He, H. Xue, W. Xiao, S. Yuen, J. Kautz, Y. Chang, U. Iqbal, L. Fan, and Y. Zhu. Sonic: Supersizing motion tracking for natural humanoid whole-body control. *Science Robotics*, 11(117):eaed4592, 2026. doi:10.1126/scirobotics.aed4592. URL https://www.science.org/doi/abs/10.1126/scirobotics.aed4592.

[2] Q. Liao, T. E. Truong, X. Huang, Y. Gao, G. Tevet, K. Sreenath, and C. K. Liu. Beyondmimic: From motion tracking to versatile humanoid control via guided diffusion. *Science Robotics*, 11(117):eadx8924, 2026. doi:10.1126/scirobotics.adx8924. URL https://www.science.org/doi/abs/10.1126/scirobotics.adx8924.

[3] R. Grandia, F. Jenelten, S. Yang, F. Farshidian, and M. Hutter. Perceptive locomotion through nonlinear model-predictive control. *IEEE Transactions on Robotics*, 39(5):3402–3421, 2023. doi:10.1109/TRO.2023.3275384.

[4] Y. Zhang, Y. Seo, J. Chen, Y. Yuan, K. Sreenath, P. Abbeel, C. Sferrazza, K. Liu, R. Duan, and G. Shi. Rpl: Learning robust humanoid perceptive locomotion on challenging terrains, 2026. URL https://arxiv.org/abs/2602.03002.

[5] Z. Wu, X. Huang, L. Yang, Y. Zhang, X. Chen, P. Abbeel, R. Duan, A. Kanazawa, C. Sferrazza, G. Shi, and C. K. Liu. Perceptive humanoid parkour: Chaining dynamic human skills via motion matching, 2026. URL https://arxiv.org/abs/2602.15827.

[6] Z. Zhuang, S. Zhu, M. Zhao, and H. Zhao. Deep whole-body parkour, 2026. URL https://arxiv.org/abs/2601.07701.

[7] A. Allshire, H. Choi, J. Zhang, D. McAllister, A. Zhang, C. M. Kim, T. Darrell, P. Abbeel, J. Malik, and A. Kanazawa. Visual imitation enables contextual humanoid control. In *Proceedings of the Conference on Robot Learning (CoRL)*, 2025.

[8] T. He, Z. Wang, H. Xue, Q. Ben, Z. Luo, W. Xiao, Y. Yuan, X. Da, F. Castañeda, S. Sastry, C. Liu, G. Shi, L. Fan, and Y. Zhu. Viral: Visual sim-to-real at scale for humanoid loco-manipulation, 2025. URL https://arxiv.org/abs/2511.15200.

[9] S. Yin, Y. Ze, H.-X. Yu, C. K. Liu, and J. Wu. Visualmimic: Visual humanoid loco-manipulation via motion tracking and generation. *arXiv preprint arXiv:2509.20322*, 2025.

[10] X. He, S. Xu, X. Li, R. Dong, L. Bian, Y.-X. Wang, and L.-Y. Gui. Ultra: Unified multimodal control for autonomous humanoid whole-body loco-manipulation. In *IROS*, 2026.

[11] J. He, C. Zhang, F. Jenelten, R. Grandia, M. Bächer, and M. Hutter. Attention-based map encoding for learning generalized legged locomotion. *Science Robotics*, 10(105):eadv3604, 2025. doi:10.1126/scirobotics.adv3604. URL https://www.science.org/doi/abs/10.1126/scirobotics.adv3604.

[12] J. Sun, G. Han, P. Sun, W. Zhao, J. Cao, J. Wang, Q. Zhang, and Y. Guo. Dpl: Depth-only perceptive humanoid locomotion via realistic depth synthesis and cross-attention terrain reconstruction. *IEEE Robotics and Automation Letters*, 11(8):9407–9414, 2026. doi:10.1109/LRA.2026.3703584.

[13] X. B. Peng, P. Abbeel, S. Levine, and M. van de Panne. Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. *ACM Transactions on Graphics*, 37 (4):1–14, 2018. doi:10.1145/3197517.3201311.

[14] Z. Qi, X. Chen, D. Liu, C. Lin, Y. Lian, S. Liang, Z. Zhang, Y. Guan, J. Wang, W. Zhang, X. Yu, H. Wang, and L. Yi. Humanoid-gpt: Scaling data and structure for zero-shot motion tracking, 2026. URL https://arxiv.org/abs/2606.03985.

[15] Z. Zhang, J. Guo, C. Chen, J. Wang, C. Lin, Y. Lian, H. Xue, Z. Wang, M. Liu, J. Lyu, H. Liu, H. Wang, and L. Yi. Track any motions under any disturbances, 2025. URL https://arxiv.org/abs/2509.13833.

[16] Z. Cheng, T. Tang, J. Lan, X. Chen, Y. Gong, Z. Liu, C. Wu, Y. Mao, Z. Deng, M. Ma, H. Xi, Y. Liu, Y. Wu, X. Wang, Y. Wang, Y. Ye, G. Huang, X. Jin, Z. Zhu, and J. Lu. Gigabrain-wbc-0.5: A behavior world model for robust whole-body control with environment interaction, 2026. URL https://arxiv.org/abs/2608.18234.

[17] 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*, 7(62):eabk2822, 2022. doi:10.1126/scirobotics.abk2822. URL https://www.science.org/doi/abs/10.1126/scirobotics.abk2822.

[18] A. Agarwal, A. Kumar, J. Malik, and D. Pathak. Legged locomotion in challenging terrains using egocentric vision. In K. Liu, D. Kulic, and J. Ichnowski, editors, *Proceedings of The 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pages 403–415. PMLR, 14–18 Dec 2023. URL https://proceedings.mlr.press/v205/agarwal23a.html.

[19] W. Sun, B. Cao, L. Chen, Y. Su, Y. Liu, Z. Xie, and H. Liu. Learning perceptive humanoid locomotion over challenging terrain, 2025. URL https://arxiv.org/abs/2503.00692.

[20] R. Singh, A. Allshire, A. Handa, N. Ratliff, and K. V. Wyk. Dextrah-rgb: Visuomotor policies to grasp anything with dexterous hands, 2025. URL https://arxiv.org/abs/2412.01791.

[21] J. Shang, K. Schmeckpeper, B. B. May, M. V. Minniti, T. Kelestemur, D. Watkins, and L. Herlant. Theia: Distilling diverse vision foundation models for robot learning. In *8th Annual Conference on Robot Learning*, 2024. URL https://openreview.net/forum?id=ylZHvlwUcI.

[22] O. Siméoni, H. V. Vo, M. Seitzer, F. Baldassarre, M. Oquab, C. Jose, V. Khalidov, M. Szafraniec, S. Yi, M. Ramamonjisoa, F. Massa, D. Haziza, L. Wehrstedt, J. Wang, T. Darcet, T. Moutakanni, L. Sentana, C. Roberts, A. Vedaldi, J. Tolan, J. Brandt, C. Couprie, J. Mairal, H. Jégou, P. Labatut, and P. Bojanowski. Dinov3, 2025. URL https://arxiv.org/abs/2508.10104.

[23] M. Tschannen, A. Gritsenko, X. Wang, M. F. Naeem, I. Alabdulmohsin, N. Parthasarathy, T. Evans, L. Beyer, Y. Xia, B. Mustafa, O. Hénaff, J. Harmsen, A. Steiner, and X. Zhai. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features, 2025. URL https://arxiv.org/abs/2502.14786.

[24] L. Pinto, M. Andrychowicz, P. Welinder, W. Zaremba, and P. Abbeel. Asymmetric actor critic for image-based robot learning. In *Proceedings of Robotics: Science and Systems*, Pittsburgh, Pennsylvania, June 2018. doi:10.15607/RSS.2018.XIV.008.

[25] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms, 2017.

[26] C. Schwarke, M. Mittal, N. Rudin, D. Hoeller, and M. Hutter. Rsl-rl: A learning library for robotics research, 2025. URL https://arxiv.org/abs/2509.10771.

[27] T. Xie, H. Zhang, J. Park, Z. Wang, B. Wen, J. Li, X. Li, Q. Ben, H. Weng, Y. Ye, D. Minor, T. Wang, C. Jiang, S. Fidler, J. Kautz, L. Fan, Y. Zhu, Z. Luo, U. Iqbal, and Y. Yuan. Grail: Generating humanoid loco-manipulation from 3d assets and video priors, 2026. URL https://arxiv.org/abs/2606.05160.

[28] L. Yang, X. Huang, Z. Wu, A. Kanazawa, P. Abbeel, C. Sferrazza, C. K. Liu, R. Duan, and G. Shi. Omniretarget: Interaction-preserving data generation for humanoid whole-body loco-manipulation and scene interaction, 2026. URL https://arxiv.org/abs/2509.26633.

[29] T. Wu, X. Kong, Y. Chen, Q. Yu, H. Ye, J. Li, Y. Wang, and H. Dong. Sugar: A scalable human-video-driven generalizable humanoid loco-manipulation learning framework, 2026. URL https://arxiv.org/abs/2605.20373.

[30] Y. Mu, Z. Zhang, Y. Shi, D. Yang, M. Matsumoto, K. Imamura, G. Tevet, C. Guo, M. Taylor, C. Shu, P. Xi, and X. B. Peng. Smp: Reusable score-matching motion priors for physics-based character control, 2025. URL https://arxiv.org/abs/2512.03028.

[31] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[32] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[33] M. Bauza, J. E. Chen, V. Dalibard, N. Gileadi, R. Hafner, M. F. Martins, J. Moore, R. Pevceviciute, A. Laurens, D. Rao, M. Zambelli, M. Riedmiller, J. Scholz, K. Bousmalis, F. Nori, and N. Heess. Demostart: Demonstration-led auto-curriculum applied to sim-to-real with multi-fingered robots. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6756–6763, 2025. doi:10.1109/ICRA55743.2025.11127813.

[34] O. M. Andrychowicz, B. Baker, M. Chociej, R. Józefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, J. Schneider, S. Sidor, J. Tobin, P. Welinder, L. Weng, and W. Zaremba. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1):3–20, 2020. doi:10.1177/0278364919887447. URL https://doi.org/10.1177/0278364919887447.

[35] K. Zakka, Q. Liao, B. Yi, L. L. Lay, K. Sreenath, and P. Abbeel. mjlab: A lightweight framework for gpu-accelerated robot learning, 2026. URL https://arxiv.org/abs/2601.22074.

[36] E. Todorov, T. Erez, and Y. Tassa. MuJoCo: A physics engine for model-based control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, 2012. doi:10.1109/IROS.2012.6386109.

[37] Google DeepMind and NVIDIA Corporation. MuJoCo Warp: A GPU-accelerated version of the MuJoCo physics simulator. GitHub repository, 2026. URL https://github.com/google-deepmind/mujoco_warp. Accessed September 8, 2026.

[38] Mustafa H. (StafaH). mjwarp-render: GPU-accelerated batched rendering for MuJoCo Warp. GitHub pull request #1113, 2026. URL https://github.com/google-deepmind/mujoco_warp/pull/1113. Merged February 6, 2026.

[39] Speedsolving.com Wiki. Sledgehammer. Online reference, 2026. URL https://www.speedsolving.com/wiki/index.php/Sledgehammer. Accessed September 8, 2026.

[40] Speedsolving.com Wiki. Sune. Online reference, 2026. URL https://www.speedsolving.com/wiki/index.php/Sune. Accessed September 8, 2026.

[41] S. Levine, C. Finn, T. Darrell, and P. Abbeel. End-to-end training of deep visuomotor policies. *J. Mach. Learn. Res.*, 17(1):1334–1373, Jan. 2016. ISSN 1532-4435.

[42] P. Ravichandar, L. Krishna, N. Sobanbabu, and Q. Nguyen. Preferenced oracle guided multi-mode policies for dynamic bipedal loco-manipulation. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6600–6606, 2025. doi:10.1109/IROS60139.2025.11246602.

[43] Z. Zhang, K. Wen, M. Xu, J. He, C. Li, T. Miki, C. Schwarke, C. Zhang, X. B. Peng, and M. Hutter. Learning whole-body humanoid locomotion via motion generation and motion tracking, 2026. URL https://arxiv.org/abs/2604.17335.