Mimic under Adversarial Perturbations (MAP)

Anirudh Shrihari
Carnegie Mellon University
ashriha2@andrew.cmu.edu

Prajwal Gurunath
Carnegie Mellon University
pgurunat@andrew.cmu.edu

Manyung Emma Hon
Carnegie Mellon University
mehon@andrew.cmu.edu

Abstract

Humanoid motion imitation methods routinely demonstrate compelling behaviors such as parkour, dancing, and running, yet these skills are typically trained for exact reproduction and evaluated only in idealized settings without real-world perturbations. Conventional reinforcement learning approaches rely on carefully engineered rewards to induce natural gait patterns, making training highly experimental and time-consuming. In this work, we study the robustness and practicality of two adversarial imitation learning methods (1) Adversarial Motion Priors (AMP) and (2) Adversarial Differential Discriminator (ADD), as a means to eliminate reward tuning while enabling motion policies that remain reliable under external perturbations. We further extend these methods to support steerability, a requirement for real-world deployment where human motion data must guide, rather than dictate, robot behavior. We introduce a comprehensive evaluation framework consisting of: (1) custom reference motions for the Unitree G1 humanoid that simulate box-carrying postures, (2) curriculum-based force-perturbation training using both constant and random external forces, and (3) an additional steering objective for ADD. Our experiments show that ADD exhibits faster and more stable learning, achieving significantly longer rollouts, while AMP produces higher motion fidelity but suffers from reduced tracking accuracy under disturbance. We discuss the trade-offs between these approaches and present strategies that enable robust, steerable, and practically deployable humanoid motion policies.

1 Introduction

Physics-based character animation and motion imitation have made remarkable progress with deep reinforcement learning [1, 2, 4]. Modern approaches can replicate complex human motions including acrobatics, dance, and athletic movements with high fidelity. However, these methods (1) are essentially replaying the motions and (2) are typically evaluated in pristine simulation environments without the perturbations common in real-world scenarios.

Real-world robotic applications demand more than just motion reproduction; they require robustness to external forces, payload handling, and environmental variations. A humanoid robot must maintain stability and natural movement while carrying objects, responding to unexpected pushes, or operating on varied terrain. This gap between impressive demonstrations and practical utility motivates our investigation.

Problem Formulation. We focus on a specific but representative challenge: training humanoid robots to accept directional commands and walk naturally while carrying a heavy item i.e, a box. This task combines several difficulties: (1) continuous external forces on the end-effectors, (2) need to maintain human-like gait patterns from reference motions, and (3) follow a steering command for the locomotion We would also like to do this with minimal reward tuning. To ensure we meet our criteria we employ and evaluate two state-of-the-art adversarial imitation learning methods: AMP [2] and ADD [4].

Contributions. Our work makes the following contributions:

- Motion retargeting from popular human motion datasets such as AMASS [9] and LAFAN [10] to the Unitree G1
- Custom dataset generation for box-carrying reference motions on the Unitree G1 humanoid
- A systematic evaluation of AMP and ADD under random external force perturbations, including large constant forces (for box carrying) and smaller random impulses (for enhanced robustness)
- Curriculum learning framework to ensure model training stability
- Implementation of steering rewards (base velocity and heading) for joint optimization of task objectives and motion imitation
- Empirical analysis revealing distinct failure modes and trade-offs: ADD's speed vs. naturalness, AMP's quality vs. tracking accuracy

2 Related Work

Physics-Based Character Control. DeepMimic [1] pioneered using deep RL for physics-based character animation by combining imitation rewards with task rewards. AMP [2] introduced adversarial motion priors, using a discriminator to distinguish policy motion from reference motion without explicit pose tracking rewards. The emphasis was on reproducing the "style" of the motions.

Differential Discriminators. ADD [4] proposed using differential observations (relative differences between policy and reference) rather than absolute states for the discriminator. This approach enables the policy to track motion patterns while adapting to external forces or different morphologies.

Robustness in Motion Imitation. While several works address domain randomization for sim-to-real transfer [8], systematic studies of external force robustness in adversarial imitation learning remain limited. FALCON [6] employs a feasible force curriculum to train humanoids for force-adaptive manipulation, but relies on extensive reward engineering rather than leveraging motion priors for natural gait patterns.

Our work addresses this gap by systematically evaluating adversarial motion imitation methods (AMP and ADD) under external force perturbations while incorporating steering commands. This approach combines natural motion quality from reference data with robustness to real-world forces, reducing the need for manual reward tuning.

3 Methods

3.1 Background: AMP and ADD

AMP uses adversarial imitation learning where a discriminator D tries to distinguish between policy-generated motions and reference motions, while the policy learns to fool the discriminator. The discriminator is trained via adversarial loss while the policy is trained via reinforcement learning with discriminator-based rewards. ADD also uses a discriminator D, but the input to the discriminator is the delta of various parameters of the policy and the reference motion. The discriminator provides a high reward if the policy motion is close to or the same as the reference motion.

AMP [2] uses absolute state observations for discrimination. The discriminator $D_\theta : \mathcal{S} \to [0, 1]$ is trained to distinguish policy states from reference states using binary cross-entropy loss. The style reward is computed as:

$$r_{style}(s_t) = -\log(1 - D_\theta(s_t)) \tag{1}$$

where s_t contains joint positions, velocities, and root information from the policy at time t. This reward encourages the policy to generate states that fool the discriminator (achieving $D_\theta(s_t) \to 1$).

ADD [4] uses differential observations computed as the difference between policy and reference states. The discriminator observes:

$$\mathbf{o}_t^{diff} = \phi(s_t^\pi, s_t^{ref}) \tag{2}$$

where ϕ computes relative features including root position difference $\Delta\mathbf{p}_{root}$, root velocity difference $\Delta\mathbf{v}_{root}$, and relative body positions with respect to the root. The discriminator $D_\theta : \mathcal{O}^{diff} \to [0, 1]$

is trained with the zero vector $\mathbf{0}$ as the only positive sample (representing perfect tracking) and error vectors $\mathbf{o}_t^{diff}$ from the policy as negative samples. The reward encourages minimizing the differential:

$$r_{style}(s_t^{\pi}, s_t^{ref}) = -\log(1 - D_\theta(\mathbf{o}_t^{diff})) \tag{3}$$

This differential formulation provides translation and rotation invariance, allowing policies to adapt their global position while maintaining relative motion patterns.

3.2 Force Perturbation Framework

We extended the MimicKit codebase to support configurable force perturbations:

Random Force Model. Forces are applied probabilistically:

$$\mathbf{f}_t = \mathbf{f}_{t-1} + \mathbf{f}_{new} \cdot \mathbb{I}(u < p_{force}) \tag{4}$$

$$\mathbf{f}_{new} \sim \mathcal{U}(-1, 1)^3 \odot \mathbf{s}_{force} \tag{5}$$

where p_{force} is application probability (0.2 per step), $\mathbf{s}_{force}$ is the force scale vector, and $u \sim \mathcal{U}(0, 1)$. The indicator function $\mathbb{I}(\cdot)$ ensures forces are applied stochastically.

Curriculum Learning. We gradually increase force magnitudes during training using a linear ramp:

$$\mathbf{s}_{force}(t) = \begin{cases} \mathbf{0} & \text{if } t < 3000 \\ \min\left(\mathbf{s}_{max}, \frac{t-3000}{6000} \cdot \mathbf{s}_{max}\right) & \text{otherwise} \end{cases} \tag{6}$$

where t is the training iteration and $\mathbf{s}_{max}$ is the target maximum force vector. Forces remain at zero for the first 3000 iterations to allow initial policy learning, then linearly increase to $\mathbf{s}_{max}$ over the next 6000 iterations, reaching maximum force at iteration 9000. The min operator is applied element-wise to cap each force component at its maximum value.

3.3 Box-Carrying Dataset Generation

We created reference motions for box-carrying by:

1. Retargeted human walking motion from AMASS [9] and LAFAN [10] to Unitree G1
2. Manually filtered out walking gaits with forward and changing direction
3. Initialize shoulder, elbow, and wrist joints to a fixed box-holding pose (elbow to wrist extended forward)

This generates naturalistic reference data where the lower body exhibits natural walking motion while the upper body maintains a carrying pose. The custom initialization of the upper-body joints ensure the reference motion represents realistic box-carrying behavior.

3.4 Steering Task with ADD

Given AMP's position drift issues under force (Section 5) and slower training to ensure stability, we developed a steerable policy using ADD to enable directional control while maintaining motion style. We implemented a steering environment where the robot must follow dynamically changing target directions while maintaining natural locomotion patterns.

Dataset Preparation. We retargeted the LAFAN dataset [10] to the Unitree G1, using 6 diverse motion sets with neutral (natural) upper-body positions. This provides varied locomotion patterns including walking, running, and transitions.

Discriminator Enhancement. We augmented ADD's differential discriminator observations with velocity information, enabling the discriminator to assess motion quality in the context of current movement direction.

Task Reward. The steering task reward combines directional objectives:

$$r_{task} = w_{tar} \cdot r_{vel} + w_{face} \cdot r_{face} \tag{7}$$

where $r_{vel} = \exp(-\|\mathbf{v} - v_{tar}\mathbf{d}_{tar}\|^2)$ rewards moving at target speed v_{tar} in target direction $\mathbf{d}_{tar}$, and $r_{face} = \exp(-\|\mathbf{d}_{heading} - \mathbf{d}_{tar}\|^2)$ rewards facing the target direction. The total reward combines task and style objectives:

$$r_{total} = r_{task} + w_{style} \cdot r_{style} \tag{8}$$

The choice of ADD over AMP for steering was motivated by ADD's superior position tracking under force, making it more suitable for directional control tasks despite overcompensation tendencies in single-axis force training.

4 Experimental Setup

Environment. We use Isaac Gym [7] with the Unitree G1 humanoid (29 DoF) on flat terrain. Control frequency is 30Hz with simulation at 120Hz. Episode length is 10 seconds with early termination on falls.

Training Details. We use PPO [5] with 4096 parallel environments. Both AMP and ADD use identical network architectures and base hyperparameters, with key differences only in discriminator formulation and batch size. Networks are 2-layer fully-connected with ReLU activation: 1024 units in the first layer and 512 in the second. Training uses SGD optimizer with learning rate 1×10^{-4} and gradient clipping (max norm 1.0). PPO hyperparameters include: discount factor γ=0.99, GAE λ=0.95, clip ratio 0.2, 5 update epochs per iteration, and 32 steps per iteration. For AMP, we use batch size 4 and reduce discriminator strength (loss weight 2.0, gradient penalty 0.5) to prevent discriminator dominance. ADD uses batch size 12. Both methods use discriminator logit regularization 0.01 and weight decay 0.0001. Training runs for 20,000-25,000 iterations (2-3 billion samples, 15-20 GPU hours on an RTX 4090).

Force Curriculum. Forces gradually increase from 0 to maximum magnitude over 192,000 environment steps (6000 iterations) starting after 3000 global steps. For [10N, 10N, 30N] configuration, forces start at [0,0,0] and linearly increase to [10N, 10N, 30N]. Forces are applied to left and right wrist links, remain constant within each episode, but maximum magnitude increases across episodes as the curriculum progresses.

Force Configurations. We test these force scenarios:

- **Constant Light:** 30N downward force on wrists (simulating 3kg box)
- **Constant Heavy:** 100N downward force on wrists (simulating 10kg box)
- **Multi-Directional:** [10N, 10N, 30N] forces in x, y, z axes to reduce single-axis overcompensation

Evaluation Metrics. We measure:

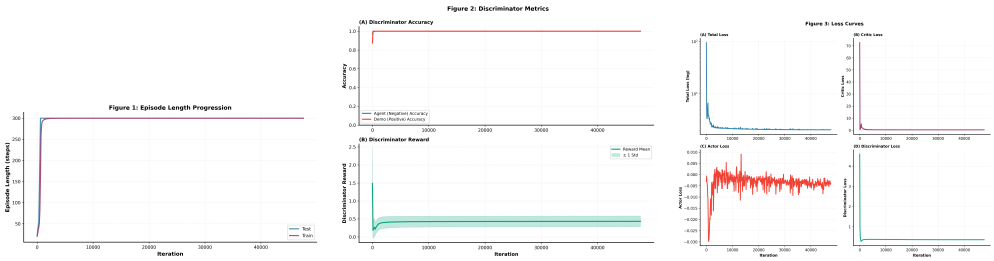
- Episode length (surrogate for stability and training progress)
- Pose tracking error: Mean joint-wise position error from reference motion
- Root tracking error: Position and velocity deviation from reference trajectory
- End-effector position error (feet and hands)
- Discriminator reward (style matching quality)

5 Results

We present our results: (1) force perturbation training on end effectors vs baselines box carrying gait with forces at test time, (2) qualitative analysis revealing the overcompensation problem, and (3) quantitative and qualitative analysis of steerable policy with ADD.

5.1 Force Perturbation Training:

We first establish a baseline by training both the ADD and AMP methods using a single walking gait reference motion, without any force-based curriculum. We then introduce force perturbation training



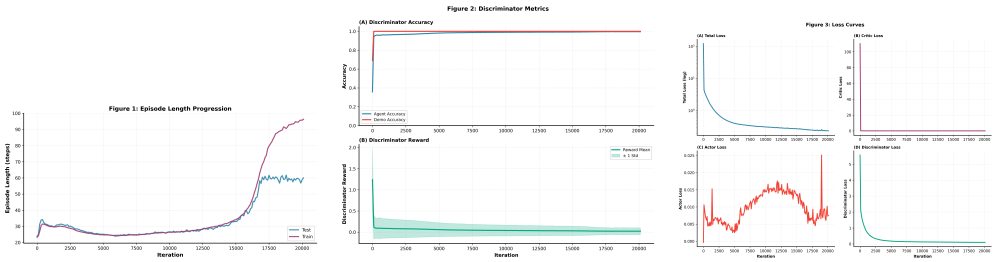
(a) ADD episode length (b) ADD discriminator (c) ADD losses

Figure 1: **ADD training curves under 30N constant force.** Episode length saturates quickly (a), discriminator loss remains high (b), but policy converges (c).

using a curriculum of external forces with magnitudes $[10\,\mathrm{N}, 10\,\mathrm{N}, 30\,\mathrm{N}]$, applied to the same walking gait. During training, we apply external forces and compare performance across multiple metrics, including velocity and position tracking, to assess the ability of the policies to counteract disturbances while preserving smooth and natural gait dynamics. Importantly, no additional force-specific reward terms are introduced during training. We also qualitatively compare the upper-body position tracking and gait at test time.

ADD Performance. Figure 1 shows ADD training curves. Key observations: (1) Episode length saturates quickly (within 5000 iterations), reaching maximum rollout duration, (2) Discriminator maintains high loss, suggesting policy motion differs from reference despite achieving stability (3) Policy learns rapid force compensation, enabling continued walking

AMP Performance. Figure 2 shows AMP training after hyperparameter tuning (gradient clipping, discriminator weakening): (1) More gradual learning curve than ADD, reaching stability around 15,000 iterations, (2) Discriminator reward steadily increases, indicating better style matching (3) Higher pose tracking accuracy but lower stability under force



(a) AMP episode length (b) AMP discriminator (c) AMP losses

Figure 2: **AMP training curves under force perturbations.** More gradual learning (a), improving discriminator reward (b), and stable loss convergence (c).

5.2 Qualitative Analysis: The Overcompensation Problem

Having established baseline training, we now investigate policy behavior under different force magnitudes to understand robustness limits.

Single-Axis Forces Cause Overcompensation in ADD Under [0,0,30N] curriculum training, ADD produces stable walking but develops unnatural compensation strategies: (1) Arms raised above head to counterbalance downward forces (2) Exaggerated leaning motions during gait (3) Persistent overcompensation: Even at zero force, policy maintains defensive posture.

The [0,0,100N] curriculum exacerbates this problem. Figure 3 shows that the aggressive curriculum creates policies that overcompensate severely even without applied forces, sometimes becoming

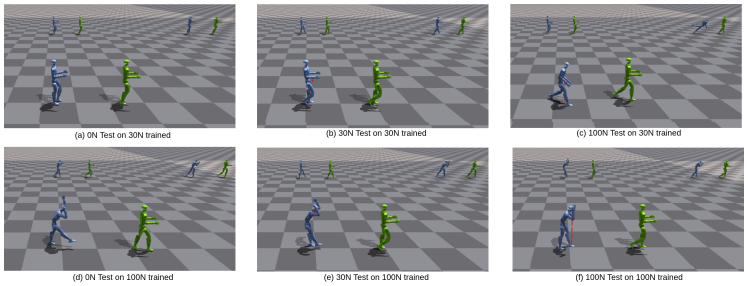


Figure 3: **ADD behavior under single-axis force curricula.** 30N curriculum allows adaptation but shows arm raising, while 100N curriculum causes severe overcompensation even at zero force.

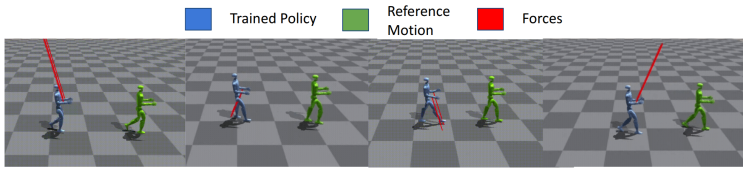


Figure 4: **Effect of multi-directional forces on ADD.** Multi-directional [10N,10N,30N] training produces natural poses.

unstable. This behavior suggests the policy learns axis-specific compensation strategies (raising arms for downward forces) that persist inappropriately.

Multi-Directional Forces Solve Overcompensation To address this issue, we trained ADD with [10N, 10N, 30N] multi-directional forces. This configuration applies forces in x, y, and z axes, preventing axis-specific compensation strategies. Figure 4 shows that multi-directional training produces natural poses even after force removal, validating our hypothesis that force diversity prevents overcompensation.

AMP: Better Pose Tracking, Lower Force Robustness AMP maintains more natural-looking poses and better pose tracking accuracy but exhibits lower force robustness. We trained AMP with the same three force configurations as ADD to enable direct comparison: (1) Significant drift in root position during walking under force, (2) Occasional stumbling and falls under moderate forces (>40N) (3) Better recovery when forces are removed compared to ADD, (4) Maintains closer match to reference motion (higher discriminator rewards)

Multi-directional forces: (1) Improved stability compared to single-axis training, (2) Still exhibits position drift, making it less suitable for directional control, (3) Maintains natural pose quality across all force magnitudes. Figure 5 shows that AMP can walk with light constant forces but experiences drift and instability with strong perturbations, confirming the trade-off between pose tracking accuracy and force robustness across all training configurations.

5.3 Quantitative Comparison

We now quantify the performance differences across force configurations and methods.

Pose Tracking and Stability Metrics Table 1 presents pose tracking error and episode length across force magnitudes for both AMP and ADD trained with all three force configurations. Key findings:

- **AMP**: Excellent pose tracking at 0N across all configs, but rapid degradation under force; multi-directional training provides modest improvement
- **ADD**: Lower pose tracking error under force but higher error at 0N with single-axis training (overcompensation)

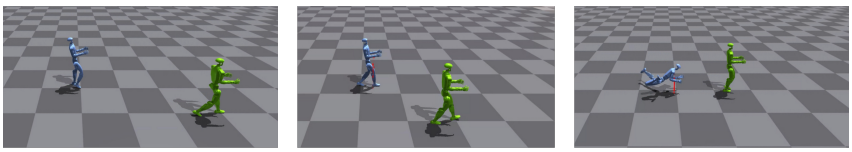


Figure 5: **AMP behavior under constant forces.** Model trained with curriculum up to 100N shows reasonable walking with light forces but drifts and falls under strong forces, suggesting a lower upper-bound when compared to ADD.

- **Multi-directional benefit**: [10,10,30] training reduces ADD's 0N pose error while maintaining robustness; AMP shows smaller improvement but still drifts.
- **Body Position Error:** ADD maintains a lower body position error with multi-directional forces due to explicitly learning and weighting the difference between the reference motion, hence the signal for tracking is more nuanced than AMP which instead learns the style of the motions
- **Episode Length:** Given that the episode ends right when the robot falls, a higher episode length results in continous tracking of repeatable motions while also countering varying forces during the episode. Multi-directional forces does allow the agent to be aware of any incoming force to compensate and continue tracking positional errors and style of the motions

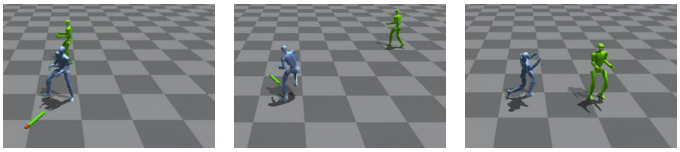
Table 1: Training results under different force curricula (final metrics after convergence)

Force Curriculum	Episode Length (s)	Body Pos Error	Iterations
ADD [0,0,30]	19.2	0.211	15000
ADD [0,0,100]	15.6	0.291	15000
ADD [10,10,30]	260.5	0.015	15000
AMP [0,0,30]	31.2	0.107	15000
AMP [0,0,100]	24.9	0.085	15000
AMP [10,10,30]	265.2	0.045	15000

Method Comparison Table 1 summarizes the key differences between AMP and ADD under force perturbations. ADD was also evaluated on steering tasks, where it demonstrated the ability to follow target directions while maintaining learned motion styles.

5.4 Qualitative Analysis: Steerable Policy

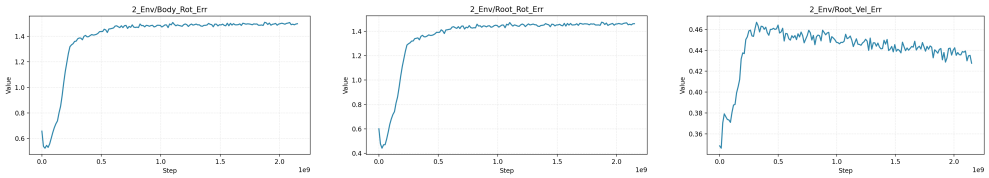
We now measure the Qualitative performance of the Steerable Policy. Figure 6 show that the policy can track the velocity, but it struggles with tracking the reference motion gait.



(a) Steerable pose 1 (b) Steerable pose 2 (c) Steerable pose 3

Figure 6: **Robot pose from steerable policy compared to reference motion**

From Figure 7, we can see that the body and root rotation error is pretty high, indicating that the policy isn't able to track the reference policy pretty well. The root velocity error is 0.4 m/s indicating that it can track velocity pretty well.



(a) Body Rotation Error (b) Root Rotation Error (c) Root Velocity Error

Figure 7: **ADD Steerable training curves**

6 Discussion

6.1 Why AMP Struggles with Pose Tracking Under Force

AMP's absolute discriminator observes full state s, requiring close pose tracking to reference for high rewards. Under external forces:

1. The policy must balance discriminator reward (match reference pose) with task reward (avoid falling under force)
2. Forces cause accumulating root position errors that degrade overall pose tracking accuracy
3. Gradient clipping and discriminator weakening (necessary for training stability) reduce corrective signals for pose tracking

The trade-off is excellent pose tracking at 0N but rapid degradation under force. AMP maintains natural motion quality but cannot sustain pose tracking accuracy under sustained perturbations.

6.2 Curriculum Learning Insights

Our force curriculum reveals important insights about force magnitude and directionality:

Force Magnitude Effects:

- **30N curriculum:** Produces policies that handle 0-40N robustly without overcompensation
- **100N curriculum:** 100N is well past the carrying capacity of the G1 and hence it tries and overcompensates it's motion

6.3 Why ADD isn't able to walk with a smooth gait while tracking velocities

The output from ADD's discriminator is very small while training. The velocity tracking reward outweighs the discriminator rewards, hence the ADD policy isn't able to learn reference motion while also tryin to track velocity. This can be fixed by weighing the rewards from ADD higher than velocity tracking, or adding a curriculum to introduce velocity tracking after the policy has learnt how to imitate the reference walking motion.

6.4 Limitations and Future Work

Current Limitations:

- Evaluation limited to downward wrist forces; real box-carrying involves distributed forces
- Flat terrain only; uneven ground would compound challenges
- No real hardware validation; sim-to-real gap remains unquantified
- Limited analysis of failure modes (what specific forces cause falls?)

Future Directions:

- **Explicit pose constraints:** Add constraints on unnatural joint configurations

- **Multi-task training:** Jointly train on steering, force perturbations, and terrain variations
- **Real robot deployment:** Evaluate on physical Unitree G1 with actual box-carrying tasks
- **Improved steerable policy:** Improve the baseline steerable ADD policy to better follow refence policy motion
- **Steerable with force compensation:** Add force curriculum to the steerable ADD policy to walk and change directions with external force perturbations.

7 Conclusion

We studied whether adversarial motion imitation can produce humanoid locomotion policies that are not only visually plausible, but also practical under real-world disturbances. Using the Unitree G1 in IsaacGym, we compared AMP and ADD under curriculum-based external force perturbations applied at the wrists to emulate box-carrying loads, and we further extended ADD with steering objectives to support velocity tracking control.

Across the force perturbation experiments, ADD consistently learned faster and achieved longer stable rollouts, indicating stronger disturbance tolerance and easier optimization. However, single-axis force curricula caused: ADD can learn brittle, axis-specific compensation strategies even when forces are removed. We showed that introducing multi-directional forces mitigates this overcompensation, improving zero-force posture while preserving robustness. In contrast, AMP tended to produce higher-fidelity motion and better pose matching in nominal conditions, but suffered from root drift and reduced stability as force magnitudes increased, highlighting a fundamental trade-off between strict style adherence and force robustness when the discriminator observes absolute states.

For steerability, our ADD-based policy successfully tracked commanded velocity targets, but exhibited large body and root rotation errors relative to the reference motion, suggesting that the task reward can dominate the style objective. Qualitatively there is a clear traccking of directional commands, but the style is diluted. This points to a clear path forward: improve reward balancing (or schedule it via curriculum), strengthen the style signal during early training, and only later introduce aggressive steering demands.

Overall, our results suggest that adversarial motion priors can substantially reduce reward engineering while enabling robust humanoid behaviors, but practical deployment requires careful perturbation design and objective scheduling. Future work will extend perturbations beyond wrist-only loads, incorporate uneven terrain and richer disturbance distributions, and validate sim-to-real transfer on hardware. We also plan to unify steering and force-robust training into a single multi-task framework to produce humanoid policies that are simultaneously natural, commandable, and resilient in realistic operating conditions.

Acknowledgments

We thank the MimicKit authors for releasing their codebase and the course staff for valuable feedback. Computational resources were provided by Carnegie Mellon University.

References

[1] Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel van de Panne. Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. *ACM Transactions on Graphics*, 37(4):1–14, 2018.

[2] Xue Bin Peng, Ze Ma, Pieter Abbeel, Sergey Levine, and Angjoo Kanazawa. Amp: Adversarial motion priors for stylized physics-based character control. *ACM Transactions on Graphics*, 40(4):1–15, 2021.

[3] Xue Bin Peng, Yunrong Guo, Lina Halper, Sergey Levine, and Sanja Fidler. Ase: Large-scale reusable adversarial skill embeddings for physically simulated characters. *ACM Transactions on Graphics*, 41(4):1–16, 2022.

[4] Ziyu Zhang, Sergey Bashkirov, Dun Yang, Yi Shi, Michael Taylor, and Xue Bin Peng. Physics-based motion imitation with adversarial differential discriminators. In *SIGGRAPH Asia 2025 Conference Papers*, 2025.

[5] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[6] Yuanhang Zhang, Yifu Yuan, Prajwal Gurunath, Ishita Gupta, Shayegan Omidshafiei, Ali-akbar Agha-mohammadi, Marcell Vazquez-Chanlatte, Liam Pedersen, Tairan He, and Guanya Shi. FALCON: Learning force-adaptive humanoid loco-manipulation. *arXiv preprint arXiv:2505.06776*, 2025.

[7] Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, and others. Isaac gym: High performance GPU-based physics simulation for robot learning. *arXiv preprint arXiv:2108.10470*, 2021.

[8] Xue Bin Peng, Marcin Andrychowicz, Wojciech Zaremba, and Pieter Abbeel. Sim-to-real transfer of robotic control with dynamics randomization. In *IEEE International Conference on Robotics and Automation*, pages 3803–3810, 2018.

[9] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *IEEE International Conference on Computer Vision (ICCV)*, pages 5442–5451, October 2019.

[10] Félix G. Harvey, Mike Yurick, Derek Nowrouzezahrai, and Christopher Pal. Robust motion in-betweening. *ACM Transactions on Graphics*, 39(4), 2020.