Heracles: Bridging Precise Tracking and Generative Synthesis for General Humanoid Control

X-Humanoid Heracles Project Team

Abstract

Achieving general-purpose humanoid control requires a delicate balance between the precise execution of commanded motions and the flexible, anthropomorphic adaptability needed to recover from unpredictable environmental perturbations. Current general controllers predominantly formulate motion control as a rigid reference-tracking problem. While effective in nominal conditions, these trackers often exhibit brittle, non-anthropomorphic failure modes under severe disturbances, lacking the generative adaptability inherent to human motor control. To overcome this limitation, we propose Heracles, a novel state-conditioned diffusion middleware that bridges precise motion tracking and generative synthesis. Rather than relying on rigid tracking paradigms or complex explicit mode-switching, Heracles operates as an intermediary layer between high-level reference motions and low-level physics trackers. By conditioning on the robot's real-time state, the diffusion model implicitly adapts its behavior: it approximates an identity map when the state closely aligns with the reference, preserving zero-shot tracking fidelity. Conversely, when encountering significant state deviations, it seamlessly transitions into a generative synthesizer to produce natural, anthropomorphic recovery trajectories. Our framework demonstrates that integrating generative priors into the control loop not only significantly enhances robustness against extreme perturbations but also elevates humanoid control from a rigid tracking paradigm to an open-ended, generative general-purpose architecture.

1. Introduction

Humanoid robots are rapidly transitioning from structured laboratory settings to complex, unstructured real-world environments. This shift demands general-purpose control architectures capable of executing precise, goal-directed motions while maintaining the flexible, anthropomorphic resilience characteristic of human motor control. In biological systems, motor behavior is rarely a rigid execution of a predefined plan; rather, humans seamlessly blend exact task execution with intuitive, generative recovery when faced with unexpected physical disturbances. Emulating this dual capability remains a formidable challenge in modern robotics.

The prevailing paradigm for general-purpose humanoid control relies heavily on reference-driven tracking. Recent advancements formulate motion control primarily as a problem of minimizing the kinematic deviation between the robot's current state and a provided reference trajectory. Powered by deep reinforcement learning, these tracking-based controllers He et al. (2025); Luo et al. (2025); Wang et al. (2026); Yin et al. (2025), such as those mimicking reference motions or utilizing universal trackers, excel in nominal conditions. They enable humanoids to faithfully replicate highly diverse sets of motion capture (MoCap) data, achieving impressive zero-shot execution for an array of highly agile and dynamic skills.

However, formulating general control strictly as a rigid tracking objective introduces a critical vulnerability: catastrophic and non-anthropomorphic failure modes under severe environmental perturbations. When a robot is pushed far from its reference trajectory, a pure tracker blindly attempts to minimize the immediate state error, often resulting in rigid, physically infeasible joint torques that lead to unnatural and unrecoverable falls. Conversely, entirely dropping the tracking objective in favor of pure generative models Li et al. (2026); Zeng et al. (2025), such as end-to-end behavior cloning, yields more natural, human-like movements but

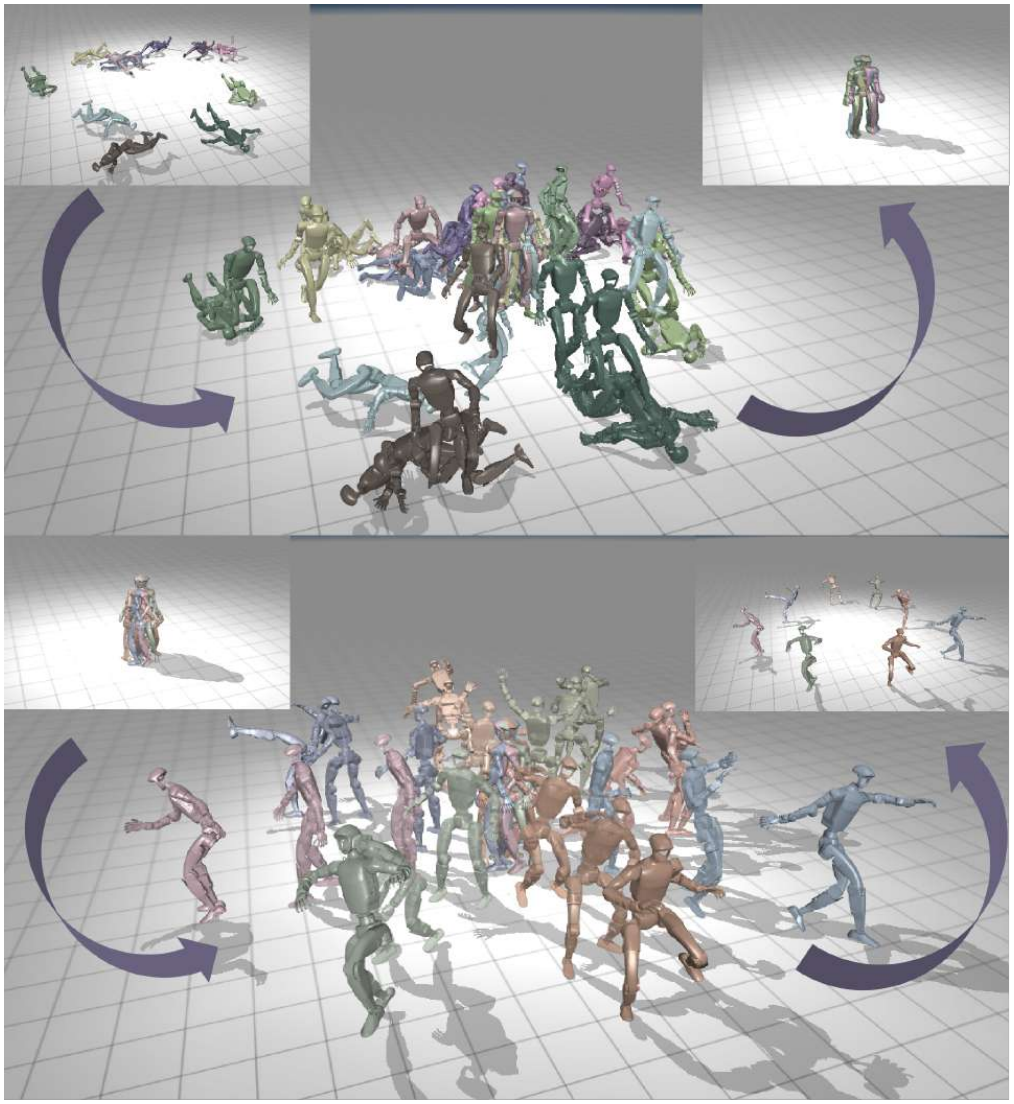

Figure 1: **Heracles synthesizes diverse, anthropomorphic recovery motions via state-conditioned diffusion.** In contrast to recovery policies trained with termination tricks that converge to a limited set of stereotypical maneuvers, Heracles leverages its generative middleware to produce a rich repertoire of agile, human-like recovery behaviors, enabling more general and robust responses across a wide range of extreme perturbations.

fundamentally sacrifices the spatial and temporal precision required for strict task execution. A significant gap remains: bridging the exactitude of precise trackers with the generative adaptability of robust human recovery.

To overcome this fundamental limitation, we propose Heracles, a novel state-conditioned diffusion middleware designed to bridge precise motion tracking and generative synthesis (Fig. 1). Rather than engineering a monolithic end-to-end controller or relying on complex, explicit state machines to switch between tracking and recovery modes, Heracles operates elegantly as an intermediary layer. Situated between the high-level original motion commands and the low-level physical execution policy, it injects powerful generative priors directly into the control loop without disrupting the high-frequency physics execution.

The core innovation of Heracles lies in its implicit, state-driven adaptability. By tightly conditioning the diffusion process on the robot's real-time physical state, the model dynamically shapes its output without explicit heuristics. When the robot's state closely aligns with the reference command, the diffusion process approximates an identity map, allowing the commands to pass through with near-zero modification to preserve strict tracking

fidelity. Conversely, when encountering significant state deviations—such as a severe push or an imminent fall—the model seamlessly transitions into a generative synthesizer. It synthesizes natural, anthropomorphic recovery trajectories that guide the underlying physics tracker back to stability. This mechanism not only enhances physical robustness but elevates humanoid control from a rigid tracking paradigm to an open-ended, generative architecture.

In summary, the primary contributions of this work are three-fold:

- **A Generative Control Middleware Paradigm:** We introduce Heracles, a novel state-conditioned diffusion middleware that uniquely bridges the precision of motion tracking with the flexibility of generative synthesis, effectively decoupling high-level intent generation from low-level physical execution.
- **Enhanced Architecture for General-Purpose Control:** We improve the underlying physics tracker and the overall control framework to better serve general-purpose tasks. This optimized architecture ensures the retention of high-fidelity tracking characteristics while seamlessly integrating with the generative priors from the middleware.
- **Robust Anthropomorphic Recovery and Motion Generalization:** We successfully deploy the proposed framework on physical humanoid robots. Extensive hardware experiments demonstrate emergent, human-like recovery behaviors and robust generative adaptability under severe, out-of-distribution physical disturbances.

2. Related Work

2.1. General Humanoid Motion Controller

Recent advances in deep reinforcement learning catalyze the development of general-purpose controllers for high-degree-of-freedom humanoid robots. These architectures aim to provide a unified execution policy for diverse motor behaviors, predominantly dividing into mimic-based reference trackers and unsupervised reinforcement learning (URL) methods.

Mimic-Based Motion Trackers. A dominant paradigm formulates motor behavior execution as a high-fidelity motion imitation task. Pioneered by DeepMimic Peng et al. (2018), which demonstrated that deep reinforcement learning can train physics-based characters to closely imitate reference motion clips, this paradigm has since been scaled to general-purpose humanoid control. Foundational large-scale frameworks, including the Generalized Motion Tracker (GMT) Chen et al. (2025), demonstrate that adaptive sampling and mixture-of-experts (MoE) architectures enable humanoids to track diverse motions via a single unified policy. Extending this paradigm, recent efforts structurally scale these models. SONIC Luo et al. (2025) expands network capacity and datasets to over 100 million frames, establishing motion tracking as a robust scalable foundation task. To break the representation bottleneck in multi-motion RL optimization, OmniXtreme Wang et al. (2026) introduces a flow-matching policy to decouple general motor skill learning from sim-to-real physical refinement. Beyond pure kinematic tracking, researchers continuously enhance the versatility and stability of these frameworks. HOVER He et al. (2025) employs multi-mode policy distillation to consolidate specific control modes—specifically navigation and loco-manipulation—into a unified controller. Concurrently, AMS Pan et al. (2026) proposes a hybrid reward scheme combining agile human MoCap data with physically constrained synthetic balance motions. More recently, BeyondMimic Liao et al. (2025) integrates a guided diffusion mechanism directly into the tracking formulation, allowing the system to solve diverse downstream tasks via classifier guidance. Beyond pure kinematics, Sun et al. (2024) pioneer the integration of large language models with adversarial imitation learning for zero-shot task execution through quantized skill representations. RoboGhost Li et al. (2026) integrates language-grounded motion latents from a motion generator with reinforcement learning for retarget-free whole-body controller. OmniRetarget Yang et al. (2025) incorporates human-scene interaction (HSI) and human-object interaction (HOI) constraints into the motion retargeting pipeline, improving the physical plausibility of transferred motions. Along similar lines, recent

works have further advanced humanoid parkour-style locomotion Zhu et al. (2026); Zhuang et al. (2026) and contact-rich interaction tasks Lin et al. (2026). MeshMimic Zhang et al. (2026) further advances the paradigm by incorporating 3D scene reconstruction from monocular video, enabling humanoid robots to learn coupled motion-terrain interactions on complex, non-flat terrains. Despite these structural advancements, tracking-first controllers remain fundamentally anchored to their reference trajectories and typically require computationally expensive test-time guidance for generation. Under severe out-of-distribution physical perturbations, relying purely on the tracking formulation predominantly results in rigid, physically infeasible joint torques rather than seamless, real-time anthropomorphic recovery.

Unsupervised RL and Skill Discovery. Conversely, unsupervised reinforcement learning (URL) and the emerging paradigm of Behavioral Foundation Models (BFMs) seek to equip robots with a diverse repertoire of motor skills devoid of explicit reference trajectories. Recent extensive surveys on BFMs Yuan et al. (2025) delineate a clear trajectory toward leveraging large-scale pre-training to capture broad behavioral priors. Zhang et al. (2024) extended this paradigm to full-size humanoid robots by developing an adversarial motion prior framework that achieves human-comparable whole-body locomotion performance. The foundational BFM framework Zeng et al. (2025) utilizes masked online distillation alongside Conditional Variational Autoencoders (CVAEs) to model behavioral distributions flexibly from unstructured data. Pushing the boundaries of autonomous exploration, cutting-edge URL approaches establish entirely reference-free policies. BFM-Zero Li et al. (2026) leverages unsupervised RL and Forward-Backward (FB) representations to create an objective-centric, promptable latent space, enabling a single generalist policy to perform zero-shot tasks and reward inference seamlessly in the real world. Several recent works Chen et al. (2026); Luo et al. (2024); Wang et al. (2025); Yu et al. (2025); Zhang et al. (2026) leverage motion tracking as a foundational mechanism to acquire human athletic skills, enabling humanoid robots to perform highly dynamic ball sports. Because these URL-driven policies and foundation models explore the state-action space unconstrained by rigid references, they inherently exhibit remarkable physical compliance and naturalistic robustness when perturbed. However, mapping these autonomously discovered, unconstrained latent spaces to high-precision, strict-fidelity spatial tracking tasks remains a formidable challenge. They fundamentally struggle to achieve the exactitude characteristic of dedicated mimic controllers in complex, dynamic execution scenarios.

2.2. Motion Generation

Synthesizing diverse, naturalistic human movements represents a foundational pursuit within computer animation. Motion-X Lin et al. (2023) provides a large-scale multi-modal human motion dataset comprising over 81K motion sequences with unified whole-body annotations spanning face, hands, and body, while its successor Motion-X++ Zhang et al. (2025) further extends the scale and diversity by incorporating additional motion sources and richer semantic labels to support more comprehensive whole-body motion generation and understanding. Early paradigm shifts leveraged score-based generative models, with the Human Motion Diffusion Model (MDM) Tevet et al. (2023) establishing a robust transformer-based baseline for text-driven kinematic synthesis. Subsequently, researchers integrated motion generation into the Large Language Model (LLM) ecosystem. MotionGPT Jiang et al. (2023) first demonstrated that treating human motion as a foreign language and unifying motion-text tasks via discrete tokenization yields strong multi-task performance. Building upon this insight, MotionGPT-2 Wang et al. (2024) further quantizes multimodal inputs—including text and single-frame poses—into LLM-interpretable tokens for unified generation and understanding. Meanwhile, the MotionMillion framework Fan et al. (2025) demonstrates that million-scale high-quality datasets coupled with autoregressive architectures unlock unprecedented zero-shot capabilities. Expanding modality fusion, OmniMotion Li et al. (2025) utilizes a continuous masked autoregressive transformer to seamlessly integrate text, speech, and music into a cohesive whole-body generation mechanism. Pushing model capacity limits, HY-Motion 1.0 Wen et al. (2025) successfully scales diffusion transformer-based flow matching models to the billion-parameter regime, yielding instruction-following digital animations with unparalleled fidelity. Despite achieving remarkable anthropomorphism and diversity, these generative foundation models fundamentally

operate in an open-loop, pure kinematic domain. They synthesize trajectories comprising joint angles and global translations completely devoid of physical constraints. Implementing these raw kinematic outputs directly on a physical humanoid invariably triggers dynamics mismatches, leading to instability and falls, because the generative process ignores crucial embodied parameters including joint torque limits, contact friction, and center-of-mass dynamics.

2.3. Hybrid Architectures for Motion Tracking and Synthesis

To overcome the inherent limitations of isolated physical trackers and open-loop generative models, recent research actively constructs hybrid architectures integrating generative synthesis with tracking objectives. Within the kinematic domain, frameworks attempt to constrain generation via tracking formulations; COMET Lee et al. (2025) employs a conditional VAE framework with a reference-guided feedback loop to prevent long-term motion degradation, while MotionStreamer Xiao et al. (2025) and DART Zhao et al. (2024) enforce sequential synthesis driven by rigorous spatial constraints. Transitioning to physically simulated characters, building upon the foundational reference-tracking paradigm of DeepMimic Peng et al. (2018), Adversarial Skill Embeddings (ASE) Peng et al. (2022) constructs continuous latent spaces from large-scale unstructured motion data, enabling characters to maintain highly anthropomorphic behaviors during diverse downstream tasks. Advancing this trajectory, the Versatile Motion Priors (VMP) framework Serifi et al. (2024) optimizes the robust control of physical characters by distilling multipurpose motion priors through a two-stage variational approach, significantly enhancing both reference trajectory tracking and resilience against external perturbations. Recent investigations further deepen this paradigm: AMOR Alegre et al. (2025) proposes multi-objective reinforcement learning to train weight-conditioned policies spanning the Pareto front of reward trade-offs, while adversarial differential discriminators Zhang et al. (2025) eliminate the need for manually-designed reward functions in physics-based motion imitation. In the multi-agent competitive domain, RoboStriker Yin et al. (2026) constructs a hierarchical framework that decouples high-level strategic reasoning from low-level physical execution via topologically constrained latent manifolds, demonstrating emergent boxing behaviors with sim-to-real transfer. However, these hybrid control paradigms retain structural deficiencies when confronting extreme, out-of-distribution physical disturbances. They typically couple high-level generative priors with low-level execution policies loosely, preventing high-frequency physical state deviations from reshaping the generative target in real time. Consequently, when encountering severe imbalance, the system fails to transition implicitly from nominal tracking to generative recovery, often reverting to rigid, explicit state-switching mechanisms. Our proposed Heracles framework resolves this bottleneck through a state-conditioned diffusion middleware that dynamically modulates the generative output based on real-time state deviations, achieving a seamless unification of precise zero-shot tracking and anthropomorphic generative recovery within a closed control loop.

3. Method

3.1. System Overview and Problem Formulation

General humanoid control requires executing desired motion commands while maintaining physical balance against unpredictable environmental disturbances. We formulate this dual objective—high-fidelity motion tracking and robust physical recovery—within a hierarchical control architecture. The proposed framework, Heracles, intrinsically decouples high-level intent generation from high-frequency physical execution (Fig. 2). It comprises two primary components: a state-conditioned generative middleware and a low-level, general-purpose physics tracking policy.

In standard tracking paradigms Chen et al. (2025); Luo et al. (2025); Peng et al. (2018), a policy minimizes the kinematic deviation between the robot's real-time proprioceptive state $\mathbf{p}_t$ and a reference motion command $\mathbf{m}_t$. When the state remains close to the reference manifold, directly tracking $\mathbf{m}_t$ yields optimal zero-shot performance. However, when severe physical perturbations push the state into out-of-distribution (OOD)

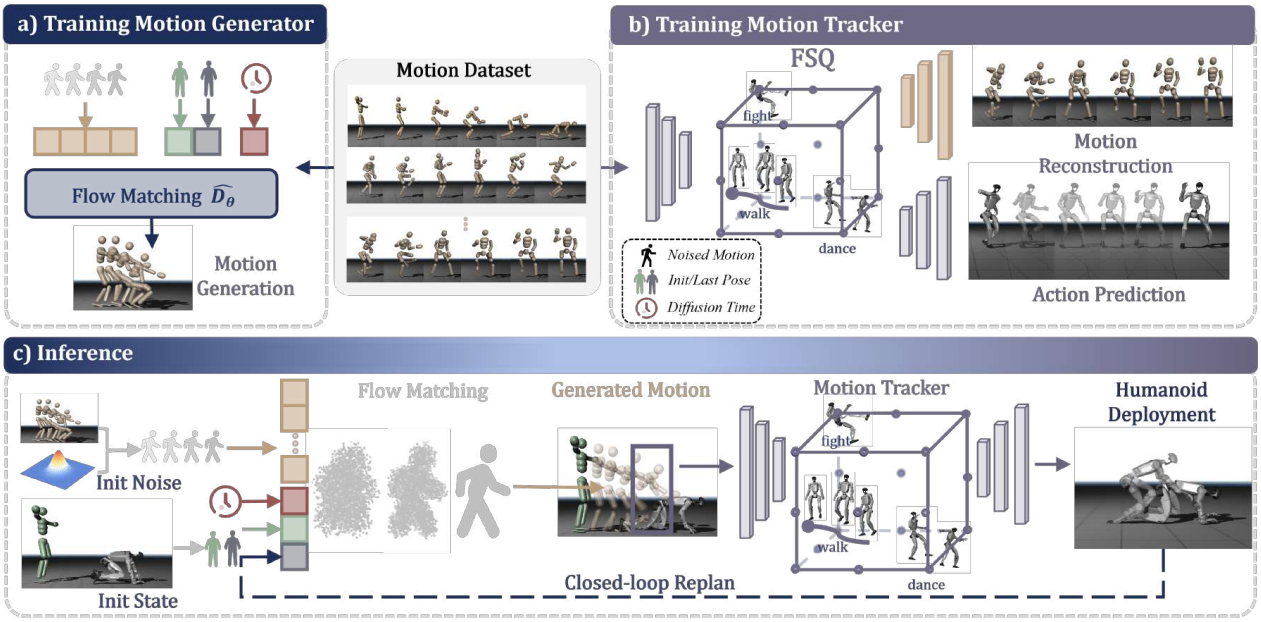

Figure 2: **Overview of the Heracles framework. (a)** A flow matching model $\hat{D}_\theta$ learns to synthesize feasible keyframe trajectories conditioned on the current state. **(b)** Reference motions are quantized into discrete tokens via FSQ, shared by reconstruction and action prediction heads. **(c)** At inference, the middleware generates trajectories through closed-loop replanning for the motion tracker to execute.

regions, forcing strict adherence to $\mathbf{m}_t$ produces rigid, myopic corrective actions that lack the coordinated multi-step reasoning required for physical recovery, invariably precipitating catastrophic failure.

To resolve this limitation, we formulate the core problem as learning an intermediary mapping that dynamically modulates the reference command based on real-time physical feasibility. The generative middleware functions as this state-conditioned trajectory synthesizer. Operating at a lower planning frequency, it observes the current state $\mathbf{p}_t$ and the original reference $\mathbf{m}_t$, predicting a short-horizon, dynamically feasible keyframe trajectory:

$$\boldsymbol{\tau}_t = f_\theta(\mathbf{p}_t, \mathbf{m}_t) \in \mathbb{R}^{K \times D}, \tag{1}$$

where K denotes the number of keyframes and D represents the state dimension. This mapping design unifies nominal tracking and OOD recovery: when the robot operates near the reference manifold, f_θ approximates an identity transformation to preserve high-fidelity tracking; conversely, under large disturbances, f_θ synthesizes entirely new, physically feasible transition trajectories that guide the robot back toward the reference manifold.

The synthesized keyframes $\boldsymbol{\tau}_t$ are subsequently densified and written into a reference buffer consumed by the low-level physics tracker. We model this continuous tracking process as a discounted Markov decision process (MDP) defined by the tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma)$. At each high-frequency control step, the tracking policy π receives an observation:

$$\mathbf{o}_t = \{\mathbf{p}_t,\ \mathbf{m}'_t,\ \mathbf{z}_d\}, \tag{2}$$

where $\mathbf{m}'_t$ denotes the modulated reference commands sampled from the densified $\boldsymbol{\tau}_t$, and $\mathbf{z}_d$ constitutes a high-level motion embedding (detailed in Sec. 3.3). The policy outputs joint-level actions $a_t \in \mathcal{A}$ to maximize the expected discounted return:

$$J(\pi) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{T-1} \gamma^t r_t\right], \tag{3}$$

where r_t represents a task reward composed of tracking precision and physical regularization terms. Deployment follows a receding-horizon replanning loop. Every N_{exec} control steps, the middleware updates $\boldsymbol{\tau}_t$ from the

latest proprioceptive observation, while the tracking policy executes the dense trajectory at the fundamental control frequency, forming a seamless, closed-loop tracking-generation architecture.

3.2. State-Conditioned Generative Middleware via Flow Matching

We formulate the trajectory generation process as a conditional flow matching problem over a geometrically constrained residual space, bridging exact motion tracking and generative synthesis without relying on explicit mode-switching heuristics.

Geometric Residual Parameterization. Directly predicting absolute state coordinates wastes model capacity on approximating the identity mapping during near-nominal execution. Instead, we predict residual trajectories relative to the current state. Let $\boldsymbol{\beta}_t$ denote the static baseline anchored at the current proprioceptive state:

$$\boldsymbol{\beta}_{t,k} = \mathbf{p}_t, \qquad \forall\, k \in \{0, \dots, K-1\}. \tag{4}$$

The middleware predicts only the residual deviation $\mathbf{r}_t$, recovering the final trajectory as:

$$\boldsymbol{\tau}_t = \boldsymbol{\beta}_t + \mathbf{r}_t. \tag{5}$$

Under this parameterization, the residual directly encodes the motion increment from the current state over the planning horizon. When the robot closely follows the reference manifold ($\mathbf{p}_t \approx \mathbf{m}_t$), the synthesized trajectory closely reproduces the original reference motion, and the middleware effectively acts as an identity map on the command signal. Under severe deviations, the model leverages the conditioning gap between $\mathbf{p}_t$ and $\mathbf{m}_t$ to synthesize recovery trajectories that diverge from the reference in favor of physical plausibility. Crucially, the target command $\mathbf{m}_t$ enters exclusively through the conditioning vector, ensuring that the residual prediction target remains independent of the state-command distance.

Continuous Conditional Flow Matching. To synthesize the complex, multimodal distributions of these recovery residuals, we employ continuous flow matching Lipman et al. (2023). Let $\mathbf{x}_0$ denote the normalized ground-truth residual data and $\mathbf{x}_1 \sim \mathcal{N}(0, \mathbf{I})$ represent the prior Gaussian noise. We define a probability path via linear interpolation:

$$\mathbf{x}_t = (1-t)\mathbf{x}_0 + t\mathbf{x}_1, \qquad t \in [0, 1]. \tag{6}$$

The model is trained to regress the underlying continuous vector field by minimizing the velocity matching objective:

$$\mathcal{L}_{\text{vel}} = \mathbb{E}_{t,\mathbf{x}_0,\mathbf{x}_1}\left[\left\|\hat{\mathbf{v}}(\mathbf{x}_t, t, \mathbf{c}_t) - (\mathbf{x}_1 - \mathbf{x}_0)\right\|_2^2\right], \tag{7}$$

where $\mathbf{c}_t = [\mathbf{p}_t, \mathbf{m}_t]$ serves as the strict state-conditioning vector. During inference, physically viable recovery trajectories are sampled by integrating the learned velocity field $\hat{\mathbf{v}}$ from $t = 1$ to $t = 0$ utilizing a minimal number of Euler steps.

Architecture and Kinematic Continuity. The velocity field is parameterized by an AdaLN-modulated Transformer Peebles and Xie (2023), where the conditioning vector $\mathbf{c}_t$ and the flow timestep embedding are injected into each block via adaptive shift-scale-gate modulation. To guarantee kinematic continuity between the real-time physical state and the synthesized trajectory, we pin the first residual token via an inpainting constraint during integration:

$$\mathbf{x}_t[0] = (1 - t_{\text{next}})\mathbf{r}_0 + t_{\text{next}}\boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}), \tag{8}$$

where $\mathbf{r}_0$ strictly defines the zero-residual anchor corresponding to the initial state.

Receding-Horizon Planning with Directional Warm Start. A key design principle is that the middleware always predicts a fixed temporal window of motion, regardless of the distance to the target command. The target is set to the current reference frame $\mathbf{m}_t$, and both the temporal window Δt and execution interval N_{exec} are held constant. Whether the robot is closely tracking the reference or operating far from the reference

manifold, the generated keyframes consistently encode the next Δt seconds of motion. Long-horizon recovery emerges autoregressively through successive replan cycles, rather than requiring the model to reason about the full trajectory in a single pass.

During inference, we initialize the ODE solver with a directional motion prior rather than pure Gaussian noise. We construct an initial residual trajectory linearly interpolating toward the target:

$$\mathbf{r}_k^{\text{init}} = \frac{k}{K-1}(\mathbf{m}_t - \mathbf{p}_t), \qquad k \in \{0, \dots, K-1\}, \tag{9}$$

and begin ODE integration from a partially noised version of this prior at $t_{\text{start}} < 1$:

$$\mathbf{x}_{t_{\text{start}}} = (1 - t_{\text{start}})\,\text{normalize}(\mathbf{r}^{\text{init}}) + t_{\text{start}}\,\boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}), \tag{10}$$

where t_{start} controls the noise-to-prior ratio. Inspired by SDEdit Meng et al. (2022), this warm start provides the linear prior as an immediate directional estimate, while the learned velocity field refines it into a natural trajectory within a reduced number of ODE steps. The resulting K keyframes are then densified into the tracker's operating frequency via cubic spline interpolation for joint positions and spherical linear interpolation for root orientations, yielding the complete dense reference signal consumed by the tracking policy.

3.3. General-Purpose Physics Tracker

3.3.1. Motion Tracking Formulation

Observation Space. The observation space is formally defined as a composite vector $\mathbf{o}_t = \{\mathbf{p}_t, \mathbf{m}_t, \mathbf{z}_d\}$, comprising the robot's proprioceptive state $\mathbf{p}_t$, the kinematic motion reference trajectory $\mathbf{m}_t$, and the discrete high-level motion embedding $\mathbf{z}_d$. Specifically, the proprioceptive state $\mathbf{p}_t$ encapsulates the immediate physical condition of the robot:

$$\mathbf{p}_t = [\mathbf{g}_t^{\text{proj}}, \boldsymbol{\omega}_t, \mathbf{q}_t - \mathbf{q}_0, \dot{\mathbf{q}}_t, \mathbf{a}_{t-1}], \tag{11}$$

where $\mathbf{g}_t^{\text{proj}} \in \mathbb{R}^3$ denotes the gravity vector projected into the local root frame, $\boldsymbol{\omega}_t \in \mathbb{R}^3$ represents the root angular velocity, $\mathbf{q}_t \in \mathbb{R}^{29}$ and $\dot{\mathbf{q}}_t \in \mathbb{R}^{29}$ represent the current joint positions and velocities respectively, $\mathbf{q}_0$ defines the default nominal joint configuration, and $\mathbf{a}_{t-1} \in \mathbb{R}^{29}$ records the previous action to ensure temporal smoothness.

The reference observation $\mathbf{m}_t$ provides per-step target kinematics:

$$\mathbf{m}_t = [\mathbf{v}_t^{\text{ref}}, \boldsymbol{\omega}_t^{\text{ref}}, \mathbf{e}_t^{\text{root}}, \mathbf{q}_t^{\text{ref}}]. \tag{12}$$

During training, $\mathbf{m}_t$ is extracted directly from the motion dataset. At deployment, it is seamlessly replaced by the densified middleware output $\mathbf{m}_t'$ without any modification to the tracker architecture. Here, $\mathbf{v}_t^{\text{ref}} \in \mathbb{R}^3$ and $\boldsymbol{\omega}_t^{\text{ref}} \in \mathbb{R}^3$ denote the reference root linear and angular velocities expressed in the body frame, $\mathbf{q}_t^{\text{ref}} \in \mathbb{R}^{29}$ specifies the target joint positions. The root orientation error $\mathbf{e}_t^{\text{root}} \in \mathbb{R}^6$ is strictly parameterized using a 6D continuous rotation feature (Rot6D), computed from the first two columns of the relative rotation matrix $R_{\text{root}}^{\text{des}} R_{\text{root}}^{\top}$. The discrete motion token $\mathbf{z}_d$ captures temporally coherent, high-level motion semantics and is detailed in the subsequent policy architecture description.

Action Space. The policy π outputs target joint positions $\mathbf{a}_t \in \mathbb{R}^{29}$. Each physical joint tracks these respective targets utilizing a low-level Proportional-Derivative (PD) controller operating at high frequency, ensuring stable torque generation.

Rewards and Domain Randomization. The reward combines positive tracking terms—covering root velocities, body-link orientations, and joint-position matching—with regularization penalties on action jerks, joint-limit violations, and undesired contacts (Tab. 1a). To ensure sim-to-real transferability, we inject comprehensive

domain randomization during training, perturbing both the simulator's physical properties (friction, center-of-mass offsets) and the command-level kinematic targets (Tab. 1b).

Table 1: **Training configuration for the general-purpose physics tracker.** Left: reward function with exponential tracking terms and regularization penalties. Right: domain randomization ranges for command-level, physical, and external disturbance parameters.

(a) Reward function. Tracking: $r_i = w_i \exp\left(-\|e_i\|^2/\sigma_i^2\right)$.

Tracked Quantity	w	σ
Root Position	0.1	0.30
Root Orientation	0.5	0.40
Root Linear Vel.	1.0	0.50
Root Angular Vel.	1.0	1.00
Rel. Body Position	1.0	0.30
Rel. Body Orientation	1.0	0.40
Body Linear Vel.	1.0	1.00
Body Angular Vel.	1.0	$\sqrt{\pi}$
Regularization	w	**Penalty**
Action Rate	-0.1	$\|a_t - a_{t\text{-}1}\|^2$
Joint Limit	-10	$\sum \max(0,\, q - q_{\text{lim}})$
Undesired Contacts	-0.1	$\sum \mathbb{1}(F_c > 1)$

(b) Domain randomization ranges.

Parameter	**Range**
▲ *Command-level perturbations*	
Target Joint Pos. (rad)	± 0.01
Target Ang. Vel. (rad/s)	± 0.2
Target Root Rot. (rad)	± 0.05
Default Joint Pos. (rad)	± 0.01
● *Physical properties*	
CoM Offset (m)	x: ± 0.5; y, z: ± 0.1
Static Friction	$[0.3,\ 1.6]$
Dynamic Friction	$[0.3,\ 1.2]$
■ *External disturbances*	
Push Frequency (s)	$1.0 - 3.0$
Push Lin. Vel. (m/s)	xy: ± 0.5; z: ± 0.2
Push Ang. Vel. (rad/s)	RP: ± 0.52; Y: ± 0.78

3.3.2. General Motion Tracking Policy

Our tracking policy is built on a shared motion-latent representation with an encoder–quantizer–decoder structure. A motion encoder maps the kinematic observations $\mathbf{m}_t$ to a continuous latent $\mathbf{z}_c$, which is then discretized into tokenized codes $\mathbf{z}_d$. Two parallel heads consume $\mathbf{z}_d$: a reconstruction decoder for representation learning, and an action decoder that fuses $\mathbf{z}_d$ with the proprioceptive state $\mathbf{p}_t$ to produce control actions.

Improved Discrete Quantization. We adopt an improved Finite Scalar Quantization (iFSQ) Lin et al. (2026) to distill high-frequency kinematic signals into compact semantic tokens. Given continuous latent features $\mathbf{z}_c \in \mathbb{R}^{N \times d}$, with N the batch size and d the embedding dimension, each channel is bounded to $[-1, 1]$ and quantized into $L = 2^K + 1$ uniformly spaced levels, where the extra center level guarantees an exact zero-state. Element-wise quantization maps each dimension to an integer index $z_d \in \{0, \ldots, L-1\}$ via:

$$z_d = \text{round}\left(\frac{L-1}{2}\left(f(z_c) + 1\right)\right). \tag{13}$$

Rather than the standard $\tanh$ bounding, we employ a sigmoid-based mapping that improves bin utilization while preserving uniform quantization intervals:

$$f(x) = 2.0\,\sigma(1.6x) - 1. \tag{14}$$

We apply the straight-through estimator (stop-gradient) for the rounding operation, yielding the discrete tokens $\mathbf{z}_d$ injected into the policy observation.

Encoder-Decoder Architecture. The motion encoder ingests a 10-frame future window $\mathbf{M}_{t:t+9}$ and produces a continuous embedding $\mathbf{z}_c \in \mathbb{R}^d$. After iFSQ discretization, the reconstruction decoder maps $\mathbf{z}_d$ back to the full 10-frame motion sequence $\hat{\mathbf{M}}_{t:t+9}$, optimized via:

$$\mathcal{L}_{\text{rec}} = \frac{1}{10}\sum_{k=0}^{9} \|\hat{\mathbf{m}}_{t+k} - \mathbf{m}_{t+k}\|_2^2. \tag{15}$$

The action decoder concatenates $\mathbf{z}_d$ with a 10-step proprioceptive history $\mathbf{P}_{t-9:t}$ to produce the control action $\mathbf{a}_t$.

Adaptive Motion Sampling. Training a unified policy over a large, heterogeneous motion corpus introduces severe optimization imbalance: uniform sampling overfits to trivial locomotion while underperforming on dynamic, agile skills. Inspired by the bin-level adaptive curriculum in ZEST Sleiman et al. (2026), we design a continuous temporal-bin variant with smoothed difficulty propagation.

We partition the concatenated motion corpus into B uniformly spaced temporal bins. At each episode termination, the per-step tracking score $s_t \in [0, 1]$ is averaged over the episode to yield a difficulty estimate $d = 1 - \bar{s}$, which is accumulated into the corresponding bin via exponential moving average:

$$F_b \leftarrow \alpha\, d_b + (1 - \alpha)\, F_b, \tag{16}$$

where F_b denotes the difficulty of bin b and α controls the update rate. To prevent isolated hard bins from dominating the sampling distribution and to propagate difficulty to temporally adjacent regions, we apply a 1D kernel smoothing operation followed by an outlier cap:

$$\tilde{F}_b = \left(\mathcal{K} * \hat{F}\right)_b, \quad \hat{F}_b = \min\left(F_b,\, c \cdot \bar{F}\right), \tag{17}$$

where $\mathcal{K}$ is an exponentially decaying kernel and c bounds the maximum bin weight relative to the global mean $\bar{F}$. The final sampling distribution mixes the smoothed difficulty with a uniform baseline to ensure exploration:

$$P_b = \eta\, \tilde{F}_b + \frac{1 - \eta}{B}, \tag{18}$$

where η controls the balance between difficulty-driven and uniform sampling. This mechanism continuously steers the training distribution toward challenging temporal regions of the motion manifold, while the kernel smoothing ensures that difficulty information propagates to neighboring segments, preventing abrupt sampling discontinuities.

3.4. Training Paradigm for Unified Tracking and Generation

Dataset Construction. Training tuples for the state-conditioned middleware are generated from a diverse motion corpus using a receding-horizon sampling strategy. For each motion sequence, segment starting points are selected at regular intervals, and the temporal segment length ℓ is drawn from a log-uniform distribution:

$$\ell \sim \exp(\mathcal{U}(\log \ell_{\min},\, \log \ell_{\max}))\,, \tag{19}$$

where $\ell_{\min} = H$ is set equal to the planning horizon. From each segment, K uniformly spaced keyframes are extracted covering only the first H frames (corresponding to $\Delta t = H/\text{fps}$ seconds), regardless of the total segment length. The conditioning vector $\mathbf{c}_t = [\tilde{\mathbf{p}}_t,\, \mathbf{m}_t]$ pairs the start state with the reference command at the segment endpoint. The residual supervision is computed against the static baseline $\boldsymbol{\beta}_{t,k} = \tilde{\mathbf{p}}_t$ (Eq. (5)), so each training target encodes the motion increment over the next Δt seconds from the current state. This design ensures that (i) the model never needs to predict trajectories exceeding a fixed temporal horizon, bounding the residual magnitude regardless of the state-command gap; (ii) training naturally covers every phase of long recovery sequences, as successive starting points within the same motion yield overlapping local windows; and (iii) eliminating near-zero-length segments ($\ell < H$) removes the trivial zero-residual bias that otherwise dominates under mean-squared-error training.

Noisy-State Augmentation. Deployment introduces a systematic discrepancy between noisy physical state estimation and the clean reference commands available during training. To close this gap, we apply asymmetric start-state perturbations: only the initial proprioceptive state is corrupted with channel-wise Gaussian noise,

$$\tilde{\mathbf{p}}_t = \mathbf{p}_t + \boldsymbol{\epsilon}, \tag{20}$$

while the target reference command remains clean. The noise magnitude is scaled per channel to reflect the varying sensitivity of different state dimensions. This asymmetric augmentation mirrors the deployment scenario where the robot's proprioceptive state reflects both sensor noise and accumulated tracking errors, while the reference command is always clean, improving robustness to real-world state-command discrepancies.

Kinematics-Aware Loss Weighting. Beyond the primary velocity matching objective (Eq. (7)), we introduce a kinematics-aware weighting scheme motivated by the observation that identical joint-space errors can induce vastly different body-space deviations depending on the current kinematic configuration. Concretely, we weight each state dimension q_d by a pose-dependent Jacobian magnitude:

$$w_d(\mathbf{q}) = \sum_b \left\| \frac{\partial \mathbf{p}_b}{\partial q_d} \right\|_2^2 \approx \sum_b \left\| \frac{\mathbf{p}_b(\mathbf{q} + \delta e_d) - \mathbf{p}_b(\mathbf{q} - \delta e_d)}{2\delta} \right\|_2^2, \tag{21}$$

where $\mathbf{p}_b$ is the Cartesian position of tracked body link b. This approximates the diagonal of $\mathbf{J}^\top \mathbf{J}$ evaluated independently at each training pose and keyframe, producing a weight tensor of shape (N, K, D). This captures pose-dependent lever arm geometry: for example, a shoulder joint in a T-pose configuration commands a longer moment arm than when the arm hangs at rest, and receives a correspondingly larger gradient signal. The weights are normalized to unit mean per dimension and clamped to a minimum value to prevent zero-gradient dimensions. In practice, all weights are precomputed via differentiable forward kinematics during dataset construction and cached alongside training tuples, eliminating all online overhead.

State Representation and Model Variants. We evaluate two state parameterizations: a 38D configuration comprising joint positions (29D), root position (3D), and root orientation in 6D continuous rotation representation, and a 35D variant omitting the global root position. The 38D formulation enables the middleware to synthesize root translational commands, allowing the robot to autonomously correct global positional drift relative to the reference trajectory. The 35D variant delegates root translation entirely to the reference motion, decoupling the middleware from global localization. While this sacrifices autonomous position correction, it eliminates dependence on external positioning systems, making it directly deployable with onboard proprioception and IMU alone. Both configurations retain equivalent fall recovery and general motion tracking performance. All quantitative results reported in this work use the 38D configuration unless otherwise noted.

4. Experiments

4.1. Implementation Details

Simulation Environment. All experiments are conducted on the Unitree G1 humanoid platform, a full-size bipedal robot standing approximately 1.32 m tall with a total mass of roughly 35 kg. The robot features 29 actuated degrees of freedom spanning the torso, two 7-DoF arms, and two 6-DoF legs, all driven by proprietary electric actuators. Training is carried out in IsaacLab Mittal et al. (2025), a GPU-accelerated simulator built on NVIDIA Isaac Sim, where we instantiate 16,384 parallel environments on a single NVIDIA A100 (80 GB) GPU. The physics simulation runs at a 5 ms timestep (200 Hz) on flat ground with randomized friction coefficients (Tab. 1b), while the control policy queries observations and emits actions at 50 Hz (every 4 simulation substeps). Each action is converted to joint torques by a per-joint PD controller executing at the full 200 Hz rate. For evaluation, all policies are tested in the MuJoCo physics engine on a held-out set comprising 101 unseen motion sequences that span locomotion, dance, martial arts, daily activities, fall-and-recovery, acrobatic jumps, and discretized motion clips in which continuous reference trajectories are replaced with piecewise-constant signals consisting of static poses separated by abrupt transitions, thereby removing all smooth interpolation and testing the policy's ability to track discontinuous commands. Each sequence is rolled out for its full duration (up to 20 s).

Motion Dataset. The training corpus is curated from diverse, complementary sources, comprising selected clips from LAFAN1 Harvey et al. (2020), 100STYLE Mason et al. (2022), SnapMoGen Guo et al. (2025),

AMASS Mahmood et al. (2019), and proprietary in-house motion capture recordings. The assembled dataset spans locomotion, martial arts, dance, daily activities, fall-and-recovery sequences, and jumping motions, thereby providing broad stylistic and temporal coverage for evaluating tracking fidelity, robustness, and execution stability under heterogeneous motion commands.

Tracker Training. The general-purpose physics tracker is trained via Proximal Policy Optimization (PPO) within an asymmetric actor–critic framework, following established practices in prior whole-body tracking systems Li et al. (2026); Liao et al. (2025); Luo et al. (2025). The iFSQ encoder, reconstruction decoder, and action decoder are trained jointly end-to-end with the PPO objective, where the reconstruction loss (Eq. (15)) and the RL policy gradient share the same encoder–quantizer pathway. The adaptive motion sampling curriculum (Eq. (18)) is activated after an initial warm-up phase to allow early-stage uniform coverage. The complete reward function and domain randomization ranges are specified in Tab. 1; all remaining hyperparameters are listed in Tab. 2.

Trajectory Generator Training. The state-conditioned flow matching trajectory generator is trained offline on paired trajectory tuples extracted from the motion corpus following the dataset construction procedure described in Eq. (19). The velocity field $\hat{\mathbf{v}}$ is parameterized by an AdaLN-modulated Transformer Peebles and Xie (2023) that predicts $K{=}8$ uniformly spaced keyframes over a fixed planning horizon of $\Delta t{=}0.2\,$s. We train two state representation variants: a **38D** configuration encoding 29 joint positions, 3D root position, and 6D root orientation, and a **35D** variant that omits root position entirely. The 38D model retains full awareness of global positioning, enabling recovery toward the reference command manifold in both pose and location. The 35D model relies solely on joint encoders and an IMU for root orientation, making it directly deployable on hardware without external localization; global position tracking is delegated to the reference motion source. We optimize the velocity matching objective (Eq. (7)) jointly with the kinematics-aware loss weighting (Eq. (21)). Noisy-state augmentation (Eq. (20)) is applied throughout training with channel-wise Gaussian noise with reduced magnitudes for root pose and orientation channels. During deployment, trajectory samples are generated via 5 Euler integration steps from $t{=}0.9$ to $t{=}0$, initialized with the directional warm start (Eq. (10)) at $t_{\text{start}}{=}0.9$. The generator replans every $N_{\text{exec}}{=}2$ control steps (0.04 s), yielding a closed-loop replanning frequency of 25 Hz. Full hyperparameters for both the tracker and the trajectory generator are listed in Tab. 2.

Table 2: **Training hyperparameters for the physics tracker and the trajectory generator.** Left: PPO-based tracker training configuration. Right: flow matching trajectory generator architecture, optimization, and inference settings.

Tracker Hyperparameter	Value
Parallel environments	16,384
Rollout horizon	24 steps
Discount factor γ	0.99
GAE λ	0.95
PPO epochs / mini-batches	5 / 4
Clipping ratio ϵ	0.2
Actor learning rate	2×10^{-3}
Critic learning rate	1×10^{-3}
KL target	0.01
Entropy coefficient	0.005
Gradient clip norm	1.0
Total training iterations	∼100,000

Generator Hyperparameter	Value
Attention blocks / heads / dim	6 / 4 / 512
Conditioning injection	AdaLN ($\mathbf{c}_t{=}[\mathbf{p}_t, \mathbf{m}_t]$ + timestep)
Keyframes K / horizon Δt	8 / 0.2 s
State dimension D	38 / 35
Optimizer	AdamW ($\beta_1{=}0.9$, $\beta_2{=}0.999$)
Learning rate	1×10^{-4} (cosine decay)
Weight decay	10^{-4}
Batch size	256
Training epochs	4,000
Parameters	22.9 M
Inference ODE steps	5 (Euler, t: 0.9→0)
Warm-start t_{start}	0.9
Replan interval N_{exec}	2 steps (0.04 s, 25 Hz)

4.2. Comparisons

We evaluate eight model configurations that systematically vary four design axes—policy architecture, motion tokenizer, observation design, and generative trajectory planning—to isolate the contribution of each component. All variants share the same simulator, robot morphology, reward function, and training budget, and are evaluated

on the identical held-out motion set.

Compared Methods. All policies receive a 10-frame proprioceptive history and a 10-frame future motion reference window (frames t through $t{+}9$) unless otherwise noted. We organize the evaluated methods into two groups; detailed network architecture specifications are provided in Tab. 3.

Architecture and external baselines. **MLP** employs a standard MLP actor–critic with BeyondMimic-style (BM) observations Liao et al. (2025). **Transformer** follows the TokenHSI architecture Pan et al. (2025), encoding proprioceptive and motion inputs into per-modality tokens that are processed by a multi-head Transformer with a learnable aggregation token before an MLP action head. **VQ-VAE** replaces the iFSQ quantizer with a VQ-VAE tokenizer Sun et al. (2024) while retaining the same encoder–decoder policy structure. **SONIC** Luo et al. (2025) is reproduced from its official open-source release with the original observation design.

Heracles variants. **iFSQ$_{\mathbf{BM}}$** pairs the iFSQ tokenizer and encoder–decoder policy with BeyondMimic observations augmented by height features (BM+H), isolating the tokenizer contribution under a standard observation design. **iFSQ$_{\mathbf{+H}}$** combines the iFSQ-based policy with the proposed observation design (Sec. 3.3) including explicit height features. **iFSQ** uses the proposed observation design without height, representing the best standalone tracker configuration. **Heracles** augments the iFSQ tracker with the state-conditioned trajectory generator (Eq. (1)), constituting the full proposed system.

Evaluation Protocol. All methods are evaluated on the same held-out set of 101 motion sequences unseen during training, spanning locomotion, dance, martial arts, daily activities, and fall-and-recovery. Each sequence is rolled out for its full duration (up to 20 s). We report five metrics: (i) *Completion Rate (CR)*—the fraction of reference frames for which the policy maintains a root height error below 0.3 m and a root orientation error below 1.2 rad; (ii) *Joint Position Error*—the L_2 norm of joint-position deviations; (iii) *Root Height Error*—absolute height deviation in the world frame; (iv) *Root Orientation Error*—orientation error excluding yaw; and (v) *Root Linear Velocity Error*—velocity error in the body frame.

Table 3: **Network architecture specifications for all evaluated methods.** $[\cdot]$ denotes MLP hidden-layer widths. iFSQ$_*$ covers all iFSQ variants (iFSQ$_{\text{BM}}$, iFSQ$_{+\text{H}}$, iFSQ), which share the same network but differ in observation design.

Method	Component	Configuration
MLP	Actor	$[4096, 2048, 1024, 512, 256]$ MLP
	Critic	$[4096, 2048, 1024, 512, 256]$ MLP
Transformer	Tokenizer	$[512, 512]$ MLP $\times$ 2 modalities $\rightarrow$ 3 tokens (512-dim)
	Backbone	3-layer, 4-head, 512-dim Transformer
	Action head	$[2048, 1024, 256]$ MLP
	Critic	$[3072, 1536, 768, 512]$ MLP
VQ-VAE	Quantizer	Codebook $\lvert\mathcal{C}\rvert$=10,240, dim=512
	Policy	Enc-Dec (identical to iFSQ)
SONIC	–	Official release Luo et al. (2025); stride-5 reference sampling
iFSQ$_*$	Policy	iFSQ Enc-Dec (Sec. 3.3)
Heracles	Tracker	iFSQ Enc-Dec (Sec. 3.3)
	Traj. gen.	6-layer, 4-head, 512-dim AdaLN Transformer Peebles and Xie (2023)

Results. Tab. 4(a) summarizes the component configuration of each variant, and quantitative tracking performance is reported in Tab. 4(b). We highlight four principal findings.

Robustness. The three variants equipped with the proposed observation design and iFSQ tokenizer (iFSQ$_{+\text{H}}$, iFSQ, Heracles) consistently outperform all baselines in completion rate, achieving 87.3%, 87.2%, and 90.6% respectively. Heracles attains the highest completion rate, exceeding the best external baseline VQ-VAE (86.0%) by 4.6 percentage points and MLP (84.8%) by 5.8 points. Among external baselines, completion rates range from 79.3% (SONIC) to 86.0% (VQ-VAE). Switching from BM to the proposed observation design while keeping

Table 4: **Component configuration and quantitative comparison of all evaluated methods. (a)** Obs. column: BM = BeyondMimic-style Liao et al. (2025); † = proposed (Sec. 3.3); +H = with height features. ✓/✗ indicates presence or absence of the trajectory generator. **(b)** Completion rate measures the fraction of reference frames for which the root height error remains below 0.3 m and the root orientation error stays below 1.2 rad; tracking errors are reported for joint positions, root height, root orientation (excl. yaw), and root linear velocity. Blue cells mark the best result per metric. Colored percentages show relative change from MLP: teal = better, red = worse.

(a) Component configuration

Method	Obs.	Tokenizer	Policy	Traj. Gen.
MLP	BM	–	MLP	✗
Transformer	BM	–	Trans.	✗
VQ-VAE	BM	VQ	Enc-Dec	✗
SONIC	SONIC	–	SONIC	✗
iFSQ_{BM}	BM+H	iFSQ	Enc-Dec	✗
$\text{iFSQ}_{+\text{H}}$	†+H	iFSQ	Enc-Dec	✗
iFSQ	†	iFSQ	Enc-Dec	✗
Heracles	†	iFSQ	Enc-Dec	✓

(b) Tracking performance

Method	CR (%) ↑	Joint Err (rad) ↓	Height Err (m) ↓	Ori Err (rad) ↓	LinVel Err (m/s) ↓
MLP	84.8	**1.1572**	0.1194	0.3590	0.2230
Transformer	80.6 (−5.0%)	1.5436 (−33.4%)	0.1426 (−19.4%)	0.3410 (+5.0%)	0.2046 (+8.3%)
VQ-VAE	86.0 (+1.4%)	2.3013 (−98.9%)	0.1077 (+9.8%)	0.3675 (−2.4%)	0.2376 (−6.5%)
SONIC	79.3 (−6.5%)	1.9828 (−71.3%)	0.1402 (−17.4%)	0.3771 (−5.0%)	0.2334 (−4.7%)
iFSQ_{BM}	85.1 (+0.4%)	1.4760 (−27.5%)	0.1096 (+8.2%)	0.3474 (+3.2%)	0.2362 (−5.9%)
$\text{iFSQ}_{+\text{H}}$	87.3 (+2.9%)	1.2924 (−11.7%)	**0.0271** (+77.3%)	**0.1539** (+57.1%)	0.1709 (+23.4%)
iFSQ	87.2 (+2.8%)	1.1863 (−2.5%)	0.0955 (+20.0%)	0.3614 (−0.7%)	**0.1561** (+30.0%)
Heracles	**90.6** (+6.8%)	1.3272 (−14.7%)	0.0764 (+36.0%)	0.2728 (+24.0%)	0.2325 (−4.3%)

the iFSQ tokenizer fixed ($\text{iFSQ}_{\text{BM}} \rightarrow$ iFSQ) raises completion from 85.1% to 87.2%, confirming the role of observation design in robust tracking.

Tokenizer effectiveness. Comparing VQ-VAE and iFSQ_{BM}—which share the same encoder–decoder policy and BM-style observation design but differ in the quantizer—reveals that iFSQ reduces the joint-position error from 2.3013 to 1.4760 rad (−35.8%) while maintaining a comparable completion rate (85.1% vs. 86.0%). The dramatic reduction in tracking precision highlights the superior codebook utilization of finite scalar quantization over conventional VQ-VAE in this high-frequency control domain. *Observation design and height features.* Among the iFSQ variants, $\text{iFSQ}_{+\text{H}}$ achieves the lowest height error (0.0271 m) and orientation error (0.1539 rad) across all methods, while iFSQ attains the best linear-velocity tracking (0.1561 m/s) and a competitive joint-position error (1.1863 rad). Removing explicit height features ($\text{iFSQ}_{+\text{H}} \rightarrow$ iFSQ) yields lower joint error (1.1863 vs. 1.2924 rad) at the expense of higher height error (0.0955 vs. 0.0271 m) and markedly degraded orientation control (0.3614 vs. 0.1539 rad), confirming that the height channel is critical for vertical and orientation precision. *Trajectory generator.* Heracles achieves the highest completion rate (90.6%) among all methods—a 6.8% relative improvement over MLP and a 3.9% improvement over the standalone iFSQ tracker—while maintaining competitive tracking quality. Compared to iFSQ, the trajectory generator reduces root orientation error from 0.3614 to 0.2728 rad (24.5% relative reduction) and height error from 0.0955 to 0.0764 m (20.0% reduction), at a modest cost in joint-position error (1.3272 vs. 1.1863 rad). This indicates that the state-conditioned generative planner synthesizes spatially-aware recovery trajectories that refine both vertical and heading control, a capability absent in the reactive tracker alone.

We present qualitative sim-to-sim evaluation results on an out-of-distribution martial arts sequence in MuJoCo, as shown in Fig. 3. Among the baseline methods, MLP, Transformer, and SONIC fail to maintain balance and collapse early in the sequence, while VQ-VAE barely tracks the motion throughout. For our ablation variants,

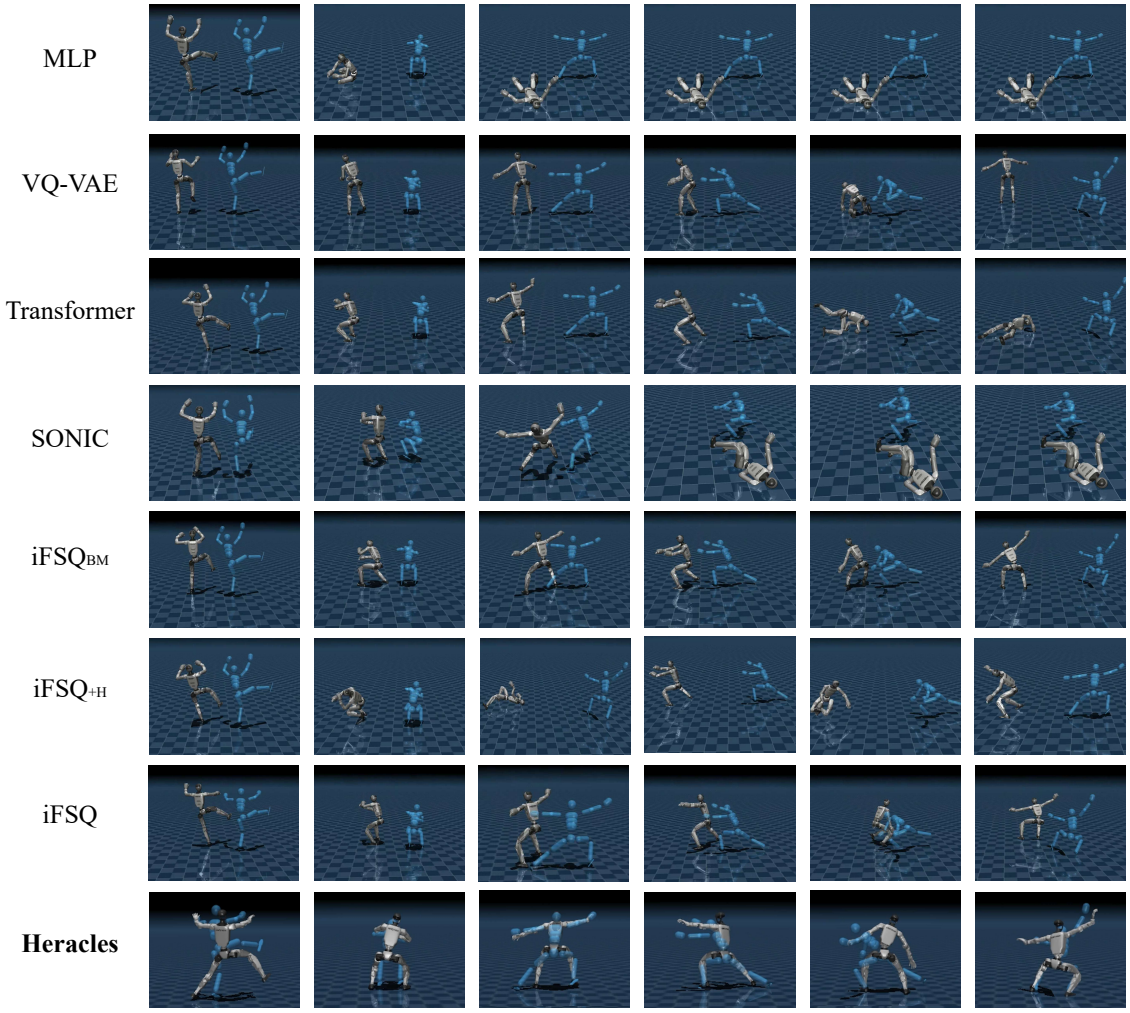

Figure 3: **Qualitative sim-to-sim comparison on an out-of-distribution martial arts sequence.** Each row shows a different method tracking the same reference motion on a Unitree G1 humanoid alongside a reference ghost in MuJoCo. MLP, Transformer, and SONIC collapse early; VQ-VAE barely tracks the motion. Among our ablations, iFSQ_{BM} and iFSQ survive without falling, while $\text{iFSQ}_{+\text{H}}$ falls but later recovers. **Heracles** (Ours) tracks the full sequence most accurately, demonstrating the strongest robustness to OOD motions.

iFSQ_{BM} and iFSQ successfully survive the entire sequence without falling; however, $\text{iFSQ}_{+\text{H}}$ experiences a fall at an intermediate stage but manages to recover afterwards. In contrast, **Heracles** not only survives the full sequence but also accurately tracks the root position and body pose across all frames, demonstrating the strongest robustness to out-of-distribution motions among all evaluated approaches.

We further validate our method through real-world deployment on a Unitree G1 humanoid robot, as illustrated in Fig. 4. The experiments span a broad spectrum of behaviors, ranging from everyday locomotion such as walking and running, to highly dynamic skills including kicking and full 360° kicks, as well as human-object interaction (HOI) scenarios.

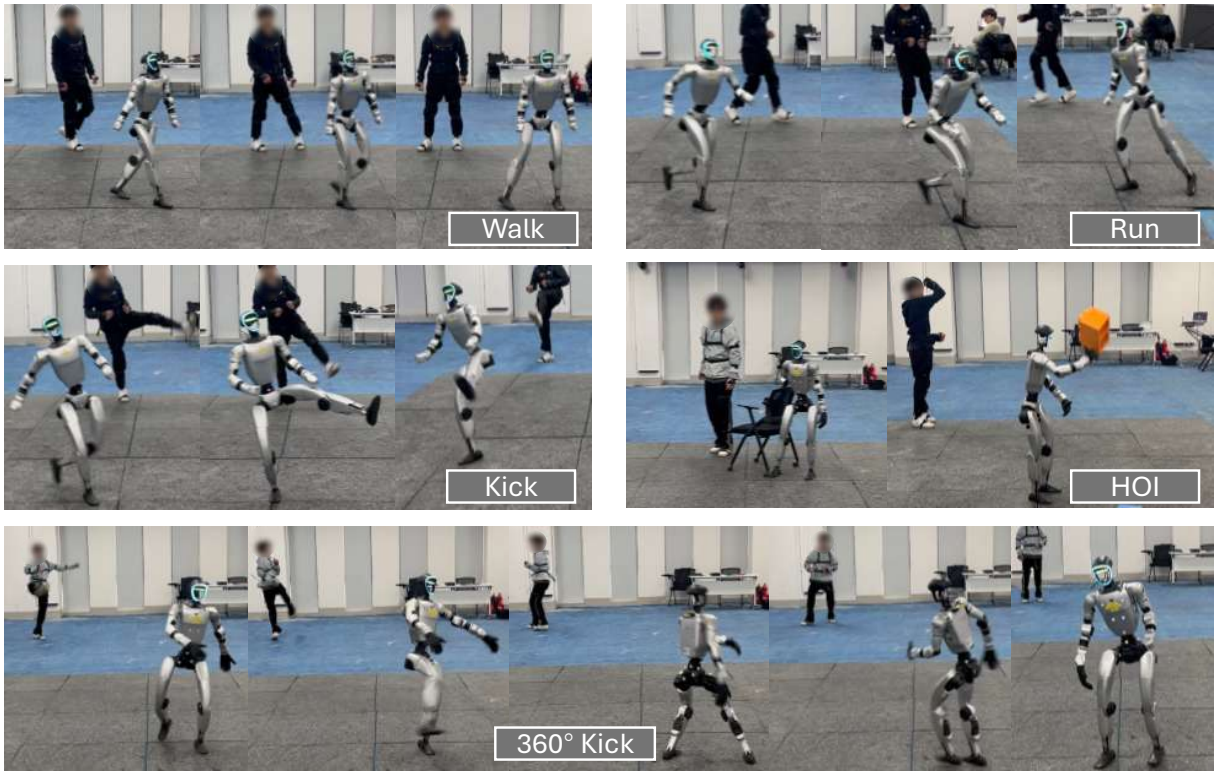

Figure 4: **Real-world motion tracking across diverse and dynamic behaviors.** Real-world experiments demonstrate that our model generalizes to a broad spectrum of motions, from everyday locomotion (walk, run) to highly dynamic skills (kick, 360° kick) and human-object interaction.

4.3. Fall-and-Recovery Evaluation

To specifically assess the fall-and-recovery capabilities that distinguish Heracles from pure tracking approaches, we conduct a dedicated evaluation on a curated subset of fall-and-recovery motion sequences extracted from the test corpus. These sequences encompass diverse recovery scenarios including lie-to-stand, prone-to-stand, and stand-to-lie transitions, with varied initial fallen configurations and recovery directions. All sequences are evaluated in both their original continuous form and discretized variants, in which the trajectories are replaced with piecewise-constant poses, to assess robustness under discontinuous reference signals. Quantitative results are summarized in Tab. 5; qualitative sim-to-sim comparisons are shown in Fig. 5.

Table 5: **Fall-and-recovery evaluation on challenging lie-to-stand, prone-to-stand, and stand-to-lie sequences.** CR denotes completion rate. Blue marks the best result per metric. Colored percentages show relative change from MLP: teal = better, red = worse.

Method	CR (%) ↑	Joint Err (rad) ↓	Height Err (m) ↓	Ori Err (rad) ↓	LinVel Err (m/s) ↓
MLP	44.0	2.1720	0.3586	1.0157	0.2710
Transformer	40.6 (−7.7%)	2.8309 (−30.3%)	0.3706 (−3.3%)	0.8355 (+17.7%)	0.2708 (+0.1%)
VQ-VAE	69.8 (+58.6%)	2.5700 (−18.3%)	0.1898 (+47.1%)	0.4307 (+57.6%)	0.2719 (−0.3%)
SONIC	42.8 (−2.7%)	2.9897 (−37.6%)	0.3342 (+6.8%)	0.8629 (+15.0%)	0.2895 (−6.8%)
iFSQ$_{BM}$	52.4 (+19.1%)	2.4482 (−12.7%)	0.2880 (+19.7%)	0.9109 (+10.3%)	0.3134 (−15.6%)
iFSQ$_{+H}$	52.7 (+19.8%)	1.7660 (+18.7%)	**0.0419** (+88.3%)	0.2744 (+73.0%)	0.2793 (−3.1%)
iFSQ	48.2 (+9.5%)	2.0236 (+6.8%)	0.3024 (+15.7%)	1.0488 (−3.3%)	**0.2405** (+11.3%)
Heracles	**90.0** (+104.5%)	**1.4114** (+35.0%)	0.0762 (+78.7%)	**0.2427** (+76.1%)	0.2830 (−4.4%)

The fall-and-recovery evaluation reveals a stark performance divide that underscores the fundamental limitations

of pure tracking paradigms (Tab. 5). All reactive tracking baselines—MLP, Transformer, and SONIC—achieve completion rates below 45%, indicating complete inability to execute fall-recovery motions. While these methods function adequately under nominal tracking conditions (Tab. 4), they fail catastrophically when the reference motion demands transitions through extreme pose configurations inherent to fall-and-recovery sequences.

Among the compared methods, VQ-VAE demonstrates unexpected resilience (CR = 69.8%), suggesting that its less precise but more flexible latent representation provides some implicit generalization to extreme poses. The iFSQ tracker variants without the generative middleware show mixed results: $\text{iFSQ}_{+\text{H}}$ achieves the highest completion rate among standalone trackers (52.7%) and the lowest height error (0.0419 m) due to its explicit height features, while iFSQ_{BM} and iFSQ achieve completion rates of 52.4% and 48.2% respectively with substantially higher tracking errors.

Heracles achieves the highest completion rate by a decisive margin (90.0%), exceeding the second-best method VQ-VAE by 20.2 percentage points—a 104.5% relative improvement over MLP. Critically, Heracles also attains the lowest joint-position error (1.4114 rad) and orientation error (0.2427 rad) among all methods, confirming that the state-conditioned generative middleware is essential for maintaining coherent tracking through the extreme state transitions characteristic of fall-and-recovery motions. By dynamically synthesizing feasible recovery trajectories conditioned on the robot's real-time physical state, Heracles bridges the gap between the reference motion and the robot's actual configuration, enabling graceful execution of motions that drive purely reactive trackers to catastrophic failure.

4.4. Ablation Studies and Architectural Analysis

To isolate the contribution of each design choice within the generative middleware, we conduct ablation experiments on the trajectory generator while keeping the iFSQ tracker fixed. All ablations are evaluated on the full 101-sequence test set spanning the complete diversity of the evaluation corpus. Results are summarized in Tab. 6.

Table 6: **Ablation study on the trajectory generator's key design components.** All variants are evaluated on the full 101-sequence test set using the same iFSQ tracker. Blue marks the best result per metric. Colored percentages show relative change from Heracles (full): teal = better, red = worse.

Variant	CR (%) ↑	Joint Err (rad) ↓	Height Err (m) ↓	Ori Err (rad) ↓	LinVel Err (m/s) ↓
Heracles (full)	**90.6**	**1.3272**	**0.0764**	**0.2728**	0.2325
w/o directional warm start	87.2 (-3.8%)	1.6236 (-22.3%)	0.0962 (-25.9%)	0.3393 (-24.4%)	0.2423 (-4.2%)
w/o noisy-state augmentation	78.6 (-13.2%)	1.8896 (-42.4%)	0.1463 (-91.5%)	0.4318 (-58.3%)	**0.2182** (+6.1%)
w/o kinematics-aware weighting	82.1 (-9.4%)	1.6931 (-27.6%)	0.1200 (-57.1%)	0.4055 (-48.6%)	0.2394 (-3.0%)

Directional Warm Start. Replacing the directional motion prior (Eq. (10)) with pure Gaussian initialization degrades all metrics: completion drops from 90.6% to 87.2% (−3.8%) and joint error increases by 22.3%. The warm start seeds the ODE solver with a coarse linear interpolation toward the target, enabling the learned velocity field to focus its refinement budget on naturalness rather than gross direction estimation. Without this prior, the generator must expend additional integration steps to discover the correct recovery heading, yielding failures particularly on fall-and-recovery sequences.

Noisy-State Augmentation. Removing the asymmetric noise injection (Eq. (20)) during training produces the most severe degradation in completion rate among all ablation variants, with CR falling to 78.6% (−13.2%) and joint error increasing by 42.4%. Height error nearly doubles (+91.5%), and orientation error increases by 58.3%. Without noise augmentation, the generator overfits to clean state inputs; at deployment, accumulated tracking drift and sensor noise push the conditioning state away from the training distribution, causing catastrophic failure on out-of-distribution motions. The augmented variant bridges this train–deploy distribution gap,

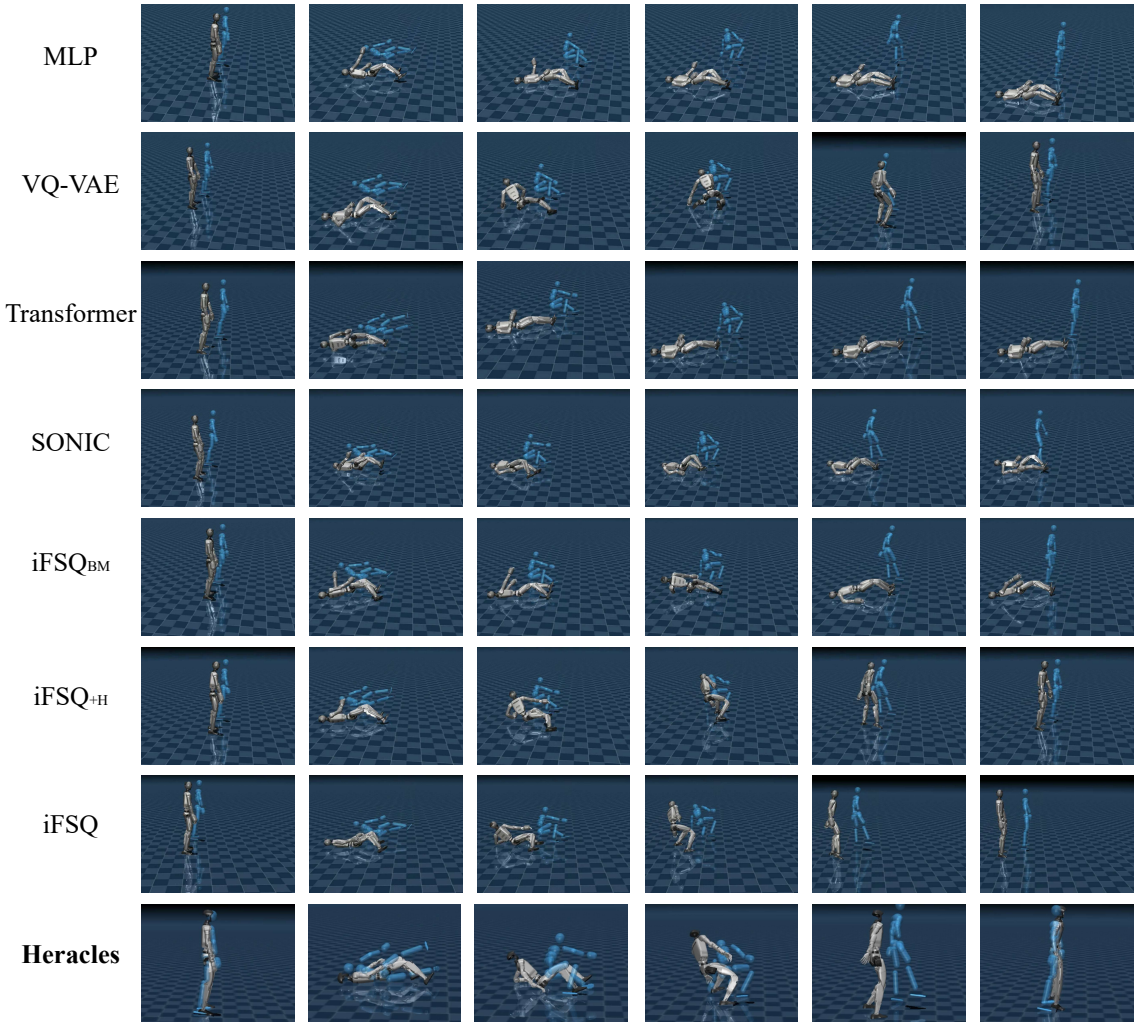

Figure 5: **Qualitative sim-to-sim comparison on an OOD lie-to-stand sequence.** Same setup as Fig. 3. MLP, Transformer, SONIC, and iFSQ_{BM} fail to stand up; VQ-VAE, $\text{iFSQ}_{+\text{H}}$, and iFSQ partially track the motion. **Heracles** (Ours) completes the full lie-to-stand transition and most accurately tracks the root position.

enabling robust trajectory generation even from noisy proprioceptive readings.

Kinematics-Aware Loss Weighting. Removing the Jacobian-based weighting (Eq. (21)) produces the largest degradation in completion rate after noisy-state augmentation, with CR falling to 82.1% (−9.4%). Height error increases by 57.1% and orientation error by 48.6%. The severity of this ablation indicates that pose-dependent lever-arm geometry is critical for robust tracking: a unit-radian shoulder error in an extended-arm configuration induces far larger Cartesian displacement than the same error with arms at rest, and the weighting scheme enables the generator to prioritize these geometrically sensitive configurations.

Learned Discrete Representation. Beyond component-level ablations, we examine whether the iFSQ tokenizer acquires a semantically structured codebook after training on the full motion corpus. Fig. 6 visualizes the discrete code activations projected into a three-dimensional embedding space, with each point representing a quantized motion token colored by its source motion category. The visualization reveals clearly separable clusters corresponding to distinct motor skills—walking, running, jumping, martial arts, dance, parkour, crawling, balance, and fall recovery—despite the quantizer receiving no explicit category labels during training. Notably, semantically related skills occupy neighboring regions (walking and running clusters lie adjacent, while crawling and balance form a separate group), indicating that the iFSQ codebook captures meaningful

kinematic similarity structure. This emergent organization confirms that the finite scalar quantization not only compresses high-frequency motion signals into compact tokens but also distills a structured motion taxonomy that enables the downstream action decoder and trajectory generator to reason over semantically coherent motion abstractions.

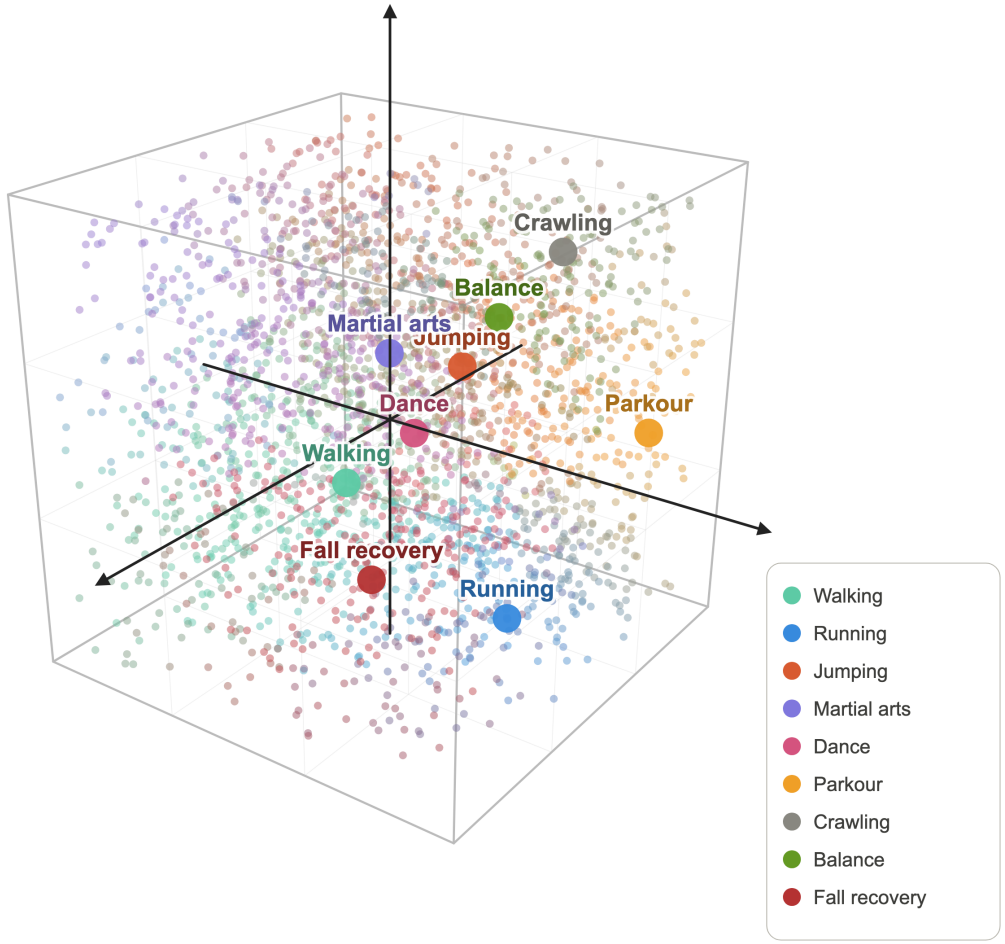

Figure 6: **Emergent semantic clustering in the learned iFSQ codebook.** Each point represents a quantized motion token projected into 3D via PCA; colors denote motion categories. Despite receiving no category labels during training, the codebook self-organizes into semantically coherent clusters corresponding to distinct motor skills.

Architectural Motivation. The ablation results collectively reveal why a hierarchical architecture—with a dedicated generative planner layered above a physics tracker—is preferable to monolithic alternatives for general-purpose humanoid control. On the 101-sequence evaluation, removing any single component reduces completion rate by 3.8–13.2%, demonstrating that all three design choices are essential for robust performance. Noisy-state augmentation has the largest impact on tracking quality (CR: -13.2%, height error: $+91.5\%$), underscoring that bridging the train–deploy distribution gap is the most critical challenge for sustained tracking fidelity. Kinematics-aware weighting produces the second-largest CR degradation (-9.4%) alongside substantial increases in height ($+57.1\%$) and orientation error ($+48.6\%$), revealing that accurate lever-arm modeling is essential for navigating complex multi-step transitions. These findings reinforce the design principle of frequency separation: the generative middleware reasons over a 0.2 s planning window at 25 Hz, synthesizing

temporally coherent recovery strategies that the tracker then faithfully executes at 50 Hz—mirroring the hierarchical structure of biological motor systems.

4.5. Recovery Behaviors and Analysis

Beyond quantitative metrics, we examine the qualitative character of recovery behaviors to understand *how* the generative middleware transforms the robot's response to severe disturbances. This analysis reveals a fundamental distinction between tracking-based and generation-based control paradigms.

Tracking-Only Failure Modes. When subjected to large external perturbations, standalone trackers—including all baselines and the iFSQ variant—exhibit a characteristic failure pattern. Upon being displaced from the reference trajectory, the tracker computes a single-step corrective action that minimizes the instantaneous state-reference error. This myopic strategy produces rigid, jerky corrective torques that lack the temporal coordination required for dynamic balance recovery. In the most severe cases, the tracker's insistence on returning to the exact reference pose forces physically infeasible joint configurations, accelerating rather than preventing the fall. Even when the tracker avoids catastrophic failure, its recovery motions appear distinctly non-anthropomorphic: abrupt whole-body stiffening, unnatural arm postures, and a conspicuous absence of the compensatory stepping strategies that characterize human balance recovery.

Generative Recovery Behaviors. Heracles produces qualitatively different recovery dynamics. When a large perturbation displaces the robot from the reference manifold, the generative middleware detects the state-reference discrepancy and synthesizes a short-horizon trajectory that prioritizes physical feasibility over immediate reference fidelity. This manifests as emergent human-like recovery strategies: compensatory stepping to widen the base of support, coordinated arm counter-motions to redistribute angular momentum, and gradual torso realignment before resuming the original motion. Crucially, these behaviors are not hand-designed or reward-engineered—they emerge naturally from the flow matching model's learned distribution over physically plausible motion transitions, conditioned on the robot's real-time state.

From Tracking to Planning: Rethinking General Humanoid Control. The observed behavioral difference reveals a deeper conceptual insight into what constitutes a truly general-purpose humanoid controller. The dominant tracking paradigm implicitly assumes that control reduces to minimizing the deviation between the robot's state and a predefined kinematic reference. While effective for nominal execution, this formulation conflates two fundamentally distinct objectives: executing a desired motor intent and maintaining physical viability.

Human motor control does not operate as a rigid reference tracker. When a person stumbles, they do not attempt to snap back to a pre-planned gait trajectory. Instead, the motor system rapidly *revises* the intended trajectory itself, generating a new plan that accounts for the current physical state, gravitational constraints, and available momentum. The original intent is temporarily deprioritized in favor of a dynamically feasible recovery path, and only once stability is restored does the system smoothly re-engage with the original task objective.

Heracles embodies precisely this principle through its state-conditioned middleware. The generative planner continuously modulates the reference signal based on real-time physical feasibility: passing commands through unmodified when tracking is viable, but seamlessly *rewriting* them when the physical state demands a different motor strategy. This transforms the controller from a passive trajectory follower into an active trajectory synthesizer that reasons about what the robot *should* do given its current physical reality, rather than what it was *told* to do by a reference signal computed without knowledge of real-time dynamics.

Implications for General-Purpose Deployment. This paradigm shift has concrete implications for deploying humanoid robots in unstructured environments. Real-world scenarios invariably introduce perturbations absent from any training distribution: unexpected collisions, terrain irregularities, payload changes, or degraded actuation. A tracking-only controller can only succeed if its training-time domain randomization happens to

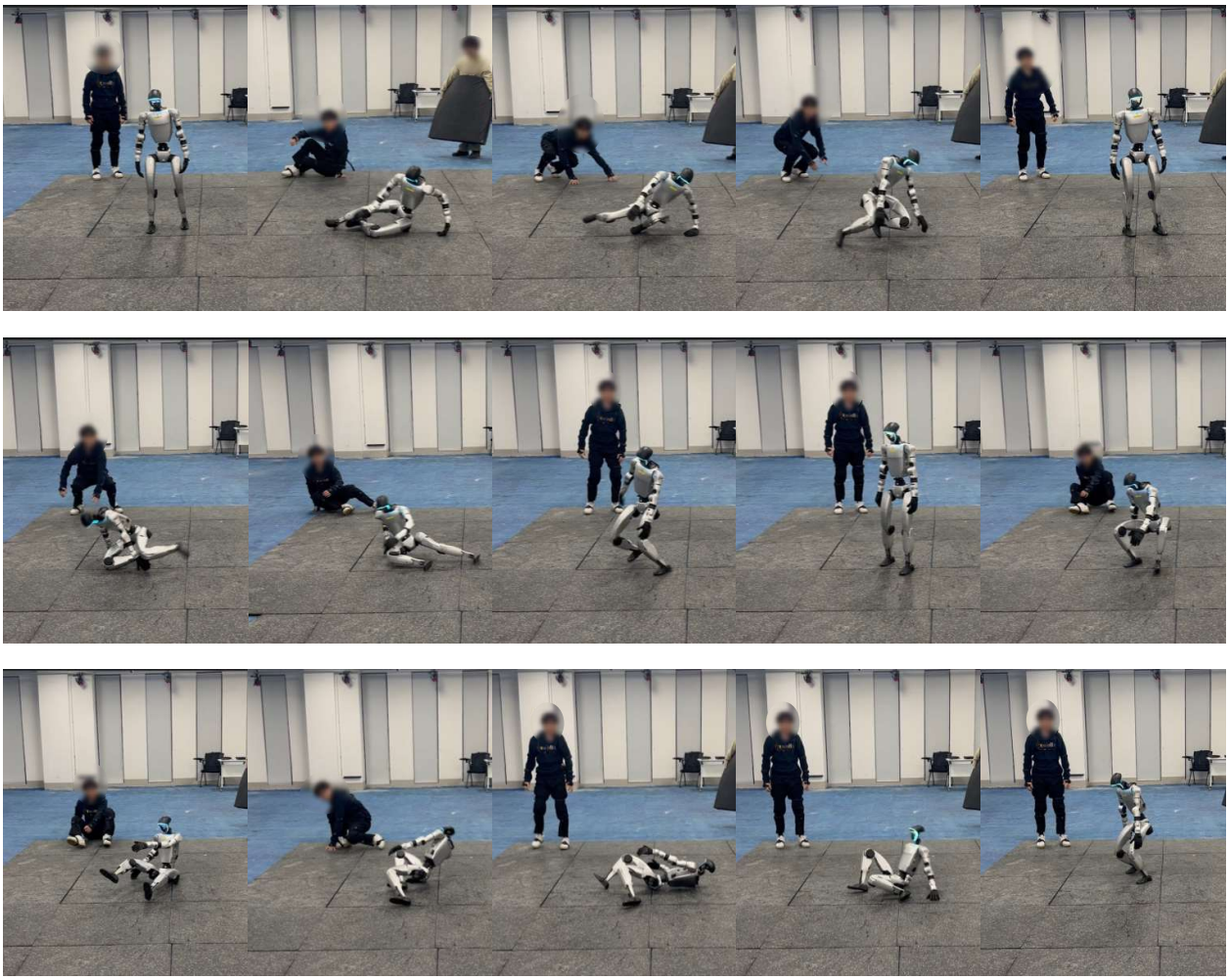

Figure 7: **Omnidirectional fall recovery motion tracking.** Real-world experiments demonstrate that our model generalizes to arbitrary lie-to-stand recovery motions, successfully handling varied initial fallen configurations and recovery directions without task-specific engineering.

cover the encountered disturbance; beyond this envelope, it fails brittly. The generative middleware, by contrast, provides a principled mechanism for *open-ended* adaptation: as long as the learned motion prior contains transitions between sufficiently diverse physical states, the model can compose novel recovery strategies for previously unseen perturbations. This compositional generalization—the ability to recombine learned motion primitives into new sequences conditioned on novel states—is what distinguishes a truly general controller from one that merely covers a large but finite set of pre-trained behaviors.

We further evaluate this generalization capability on omnidirectional fall recovery tasks in both simulation and the real world. As shown in Fig. 5, Heracles completes the full lie-to-stand transition in MuJoCo while all baseline methods fail or only partially succeed. Fig. 7 presents the corresponding real-world results: across three trials, the robot is initialized in distinct fallen configurations—supine, lateral, and prone postures. In all cases, the robot successfully executes a complete lie-to-stand recovery by following the reference motion, progressively transitioning through intermediate support phases. Critically, the recovery directions vary across trials, with the robot rising toward different orientations relative to its initial fallen heading, demonstrating true omnidirectional recovery rather than a memorized fixed-direction strategy.

5. Conclusion

In conclusion, this work presents Heracles, a state-conditioned generative middleware that fundamentally resolves the longstanding dichotomy between strict kinematic tracking and robust physical recovery in humanoid robotics. By embedding a continuous flow matching process within a closed-loop control architecture, the framework dynamically bridges high-fidelity intent execution with anthropomorphic resilience. Without relying on explicit mode-switching heuristics, Heracles intrinsically preserves exact zero-shot tracking under nominal conditions while seamlessly synthesizing dynamically feasible recovery maneuvers during severe environmental perturbations. Ultimately, this unified paradigm liberates embodied control from rigid, reference-bound execution, establishing a highly scalable foundation for deploying agile and resilient general-purpose humanoids in complex physical environments.

6. X-Humanoid Heracles Project Team

This report reflects a collaborative effort by the X-Humanoid Heracles project team. The roles and contributors are listed below.

Project Leader. Qiang Zhang

Equal Contribution. Zelin Tao, Zeran Su

Project Team Members. Peiran Liu, Jingkai Sun, Wenqiang Que, Jiahao Ma, Jialin Yu, Jiahang Cao, Pihai Sun, Hao Liang

Technical Support. Gang Han, Wen Zhao, Zhiyuan Xu, Yijie Guo, Jian Tang

References

[1] Lucas N Alegre, Agon Serifi, Ruben Grandia, David Müller, Espen Knoop, and Moritz Bächer. Amor: Adaptive character control through multi-objective reinforcement learning. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference (SIGGRAPH)*, 2025. 5

[2] Yeke Chen, Shihao Dong, Xiaoyu Ji, Jingkai Sun, Zeren Luo, Liu Zhao, Jiahui Zhang, Wanyue Li, Ji Ma, Bowen Xu, et al. Learning human-like badminton skills for humanoid robots. *arXiv preprint arXiv:2602.08370*, 2026. 4

[3] Zixuan Chen, Mazeyu Ji, Xuxin Cheng, Xuanbin Peng, Xue Bin Peng, and Xiaolong Wang. Gmt: General motion tracking for humanoid whole-body control. *arXiv preprint arXiv:2506.14770*, 2025. 3, 5

[4] Ke Fan, Shunlin Lu, Minyue Dai, Runyi Yu, Lixing Xiao, Zhiyang Dou, Junting Dong, Lizhuang Ma, and Jingbo Wang. Go to zero: Towards zero-shot motion generation with million-scale data. 2025. 4

[5] Chuan Guo, Inwoo Hwang, Jian Wang, and Bing Zhou. Snapmogen: Human motion generation from expressive texts. In *In Advances in Neural Information Processing Systems (NeurIPS)*, 2025. 11

[6] Félix G Harvey, Mike Yurick, Derek Nowrouzezahrai, and Christopher Pal. Robust motion in-betweening. *ACM Transactions on Graphics (TOG)*, 39(4):60–1, 2020. 11

[7] Tairan He, Wenli Xiao, Toru Lin, Zhengyi Luo, Zhenjia Xu, Zhenyu Jiang, Jan Kautz, Changliu Liu, Guanya Shi, Xiaolong Wang, Linxi Fan, and Yuke Zhu. Hover: Versatile neural whole-body controller for humanoid robots. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2025. 1, 3

[8] Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. Motiongpt: Human motion as a foreign language. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 4

[9] Eunjong Lee, Eunhee Kim, Sanghoon Hong, Eunho Jung, and Jihoon Kim. Controllable long-term motion generation with extended joint targets. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2025. 5

[10] Yitang Li, Zhengyi Luo, Tonghe Zhang, Cunxi Dai, Anssi Kanervisto, Andrea Tirinzoni, Haoyang Weng, Kris Kitani, Mateusz Guzek, Ahmed Touati, Alessandro Lazaric, Matteo Pirotta, and Guanya Shi. Bfm-zero: A promptable behavioral foundation model for humanoid control using unsupervised reinforcement learning. In *In International Conference on Learning Representations (ICLR)*, 2026. 1, 4, 12

[11] Zhe Li, Weihao Yuan, Weichao Shen, Siyu Zhu, Zilong Dong, and Chang Xu. Omnimotion: Multimodal motion generation with continuous masked autoregression. *arXiv preprint arXiv:2510.14954*, 2025. 4

[12] Zhe Li, Cheng Chi, Yangyang Wei, Boan Zhu, Yibo Peng, Tao Huang, Pengwei Wang, Zhongyuan Wang, Shanghang Zhang, and Chang Xu. From language to locomotion: Retargeting-free humanoid control via motion latent guidance. In *In International Conference on Learning Representations (ICLR)*, 2026. 3

[13] Qiayuan Liao, Takara E Truong, Xiaoyu Huang, Guy Tevet, Koushil Sreenath, and C Karen Liu. Beyondmimic: From motion tracking to versatile humanoid control via guided diffusion. *arXiv preprint arXiv:2508.08241*, 2025. 3, 12, 13, 14

[14] Bin Lin, Yujia Ge, Xinyang Cheng, Ye Zhu, Shubin Ye, Yuan Li, Jiaxi Zhu, Jiahao Yan, Haoqian Zeng, Zhenyu Wang, Liuhan Zhang, Fang Wan, Qingdong Liu, Xianyi Zhao, Yonghong Li, and Limin Yang. ifsq: Improving fsq for image generation with 1 line of code. *arXiv preprint arXiv:2601.17124*, 2026. 9

[15] Jing Lin, Ailing Zeng, Shunlin Lu, Yuanhao Cai, Ruimao Zhang, Haoqian Wang, and Lei Zhang. Motion-x: A large-scale 3d expressive whole-body human motion dataset. *Advances in Neural Information Processing Systems ((NeurIPS)*, 36:25268–25280, 2023. 4

[16] Yutang Lin, Jieming Cui, Yixuan Li, Baoxiong Jia, Yixin Zhu, and Siyuan Huang. Lessmimic: Long-horizon humanoid interaction with unified distance field representations. *arXiv preprint arXiv:2602.21723*, 2026. 4

[17] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023. 7

[18] Zhengyi Luo, Jiashun Wang, Kangni Liu, Haotian Zhang, Chen Tessler, Jingbo Wang, Ye Yuan, Jinkun Cao, Zihui Lin, Fengyi Wang, et al. Smplolympics: Sports environments for physically simulated humanoids. *arXiv preprint arXiv:2407.00187*, 2024. 4

[19] Zhengyi Luo, Ye Yuan, Tingwu Wang, Chenran Li, Sirui Chen, Fernando Castañeda, Zi-Ang Cao, Jiefeng Li, David Minor, Qingwei Ben, et al. Sonic: Supersizing motion tracking for natural humanoid whole-body control. *arXiv preprint arXiv:2511.07820*, 2025. 1, 3, 5, 12, 13

[20] Naureen Mahmood, Nima Ghorbani, Nikolaus F Troje, Gerard Pons-Moll, and Michael J Black. Amass: Archive of motion capture as surface shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5442–5451, 2019. 12

[21] Ian Mason, Sebastian Starke, and Taku Komura. Real-time style modelling of human locomotion via feature-wise transformations and local motion phases. *Proc. ACM Comput. Graph. Interact. Tech.*, 5(1), May 2022. doi: 10.1145/3522618. URL https://doi.org/10.1145/3522618. 11

[22] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations (ICLR)*, 2022. 8

[23] Mayank Mittal, Pascal Roth, James Tigue, Antoine Richard, Octi Zhang, Peter Du, Antonio Serrano-Muñoz, Xinjie Yao, René Zurbrügg, Nikita Rudin, et al. Isaac lab: A gpu-accelerated simulation framework for multi-modal robot learning. *arXiv preprint arXiv:2511.04831*, 2025. 11

[24] Liang Pan, Zeshi Yang, Zhiyang Dou, Wenjia Wang, Buzhen Huang, Bo Dai, Taku Komura, and Jingbo Wang. Tokenhsi: Unified synthesis of physical human-scene interactions through task tokenization. In *In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2025. 13

[25] Yixuan Pan, Ruoyi Qiao, Li Chen, Kashyap Chitta, Liang Pan, Haoguang Mai, Qingwen Bu, Hao Zhao, Cunyuan Zheng, Ping Luo, and Hongyang Li. Agility meets stability: Versatile humanoid control with heterogeneous data. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2026. 3

[26] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 7, 12, 13

[27] 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 (TOG)*, 37 (4):1–14, 2018. 3, 5

[28] 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 (TOG)*, 41(4):1–17, 2022. 5

[29] Agon Serifi, Ruben Grandia, Espen Knoop, Markus Gross, and Moritz Bächer. Vmp: Versatile motion priors for robustly tracking motion on physical characters. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA)*, pages 1–11, 2024. 5

[30] Jean-Pierre Sleiman, He Li, Alphonsus Adu-Bredu, Robin Deits, Arun Kumar, Kevin Bergamin, Mohak Bhardwaj, Scott Biddlestone, Nicola Burger, Matthew A. Estrada, Francesco Iacobelli, Twan Koolen, Alexander Lambert, Erica Lin, M. Eva Mungai, Zach Nobles, Shane Rozen-Levy, Yuyao Shi, Jiashun Wang, Jakob Welner, Fangzhou Yu, Mike Zhang, Alfred Rizzi, Jessica Hodgins, Sylvain Bertrand, Yeuhi Abe, Scott Kuindersma, and Farbod Farshidian. Zest: Zero-shot embodied skill transfer for athletic robot control. *arXiv preprint arXiv:2602.00401*, 2026. 10

[31] Jingkai Sun, Qiang Zhang, Yiqun Duan, Xiaoyang Jiang, Chong Cheng, and Renjing Xu. Prompt, plan, perform: Llm-based humanoid control via quantized imitation learning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024. 3, 13

[32] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. Human motion diffusion model. In *International Conference on Learning Representations (ICLR)*, 2023. 4

[33] Yinhuai Wang, Qihan Zhao, Runyi Yu, Hok Wai Tsui, Ailing Zeng, Jing Lin, Zhengyi Luo, Jiwen Yu, Xiu Li, Qifeng Chen, et al. Skillmimic: Learning basketball interaction skills from demonstrations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17540–17549, 2025. 4

[34] Yuan Wang, Di Huang, Yaqi Zhang, Wanli Ouyang, Jile Jiao, Xuetao Feng, Yan Zhou, Pengfei Wan, Shixiang Tang, and Dan Xu. Motiongpt-2: A general-purpose motion-language model for motion generation and understanding. *arXiv preprint arXiv:2410.21747*, 2024. 4

[35] Yunshen Wang, Shaohang Zhu, Peiyuan Zhi, Yuhan Li, Jiaxin Li, Yong-Lu Li, Yuchen Xiao, Xingxing Wang, Baoxiong Jia, and Siyuan Huang. Omnixtreme: Breaking the generality barrier in high-dynamic humanoid control. *arXiv preprint arXiv:2602.23843*, 2026. 1, 3

[36] Yuxin Wen, Qing Shuai, Di Kang, Jing Li, Cheng Wen, Yue Qian, Ningxin Jiao, Changhai Chen, Weijie Chen, Yiran Wang, et al. Hy-motion 1.0: Scaling flow matching models for text-to-motion generation. *arXiv preprint arXiv:2512.23464*, 2025. 4

[37] Lixing Xiao, Shunlin Lu, Huaijin Pi, Ke Fan, Liang Pan, Yueer Zhou, Ziyong Feng, Xiaowei Zhou, Sida Peng, and Jingbo Wang. Motionstreamer: Streaming motion generation via diffusion-based autoregressive model in causal latent space. *arXiv preprint arXiv:2503.15451*, 2025. 5

[38] Lujie Yang, Xiaoyu Huang, Zhen Wu, Angjoo Kanazawa, Pieter Abbeel, Carmelo Sferrazza, C Karen Liu, Rocky Duan, and Guanya Shi. Omniretarget: Interaction-preserving data generation for humanoid whole-body loco-manipulation and scene interaction. *arXiv preprint arXiv:2509.26633*, 2025. 3

[39] Kangning Yin, Weishuai Zeng, Ke Fan, Minyue Dai, Zirui Wang, Qiang Zhang, Zheng Tian, Jingbo Wang, Jiangmiao Pang, and Weinan Zhang. Unitracker: Learning universal whole-body motion tracker for humanoid robots. *arXiv preprint arXiv:2507.07356*, 2025. 1

[40] Kangning Yin, Zhe Cao, Wentao Dong, Weishuai Zeng, Tianyi Zhang, Qiang Zhang, Jingbo Wang, Jiangmiao Pang, Ming Zhou, and Weinan Zhang. Robostriker: Hierarchical decision-making for autonomous humanoid boxing. *arXiv preprint arXiv:2601.22517*, 2026. 5

[41] Runyi Yu, Yinhuai Wang, Qihan Zhao, Hok Wai Tsui, Jingbo Wang, Ping Tan, and Qifeng Chen. Skillmimic-v2: Learning robust and generalizable interaction skills from sparse and noisy demonstrations. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pages 1–11, 2025. 4

[42] Mingqi Yuan, Tao Yu, Wenqi Ge, Xiuyong Yao, Huijiang Wang, Jiayu Chen, Bo Li, Wei Zhang, Wenjun Zeng, Hua Chen, and Xin Jin. A survey of behavior foundation model: Next-generation whole-body control system of humanoid robots. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. 4

[43] Weishuai Zeng, Shunlin Lu, Kangning Yin, Xiaojie Niu, Minyue Dai, Jingbo Wang, and Jiangmiao Pang. Behavior foundation model for humanoid robots. *arXiv preprint arXiv:2509.13780*, 2025. 1, 4

[44] Qiang Zhang, Peter Cui, David Yan, Jingkai Sun, Yiqun Duan, Gang Han, Wen Zhao, Weining Zhang, Yijie Guo, Arthur Zhang, and Renjing Xu. Whole-body humanoid robot locomotion with human reference. *arXiv preprint arXiv:2402.18294*, 2024. 4

[45] Qiang Zhang, Jiahao Ma, Peiran Liu, Shuai Shi, Zeran Su, Zifan Wang, Jingkai Sun, Wei Cui, Jialin Yu, Gang Han, Wen Zhao, Pihai Sun, Kangning Yin, Jiaxu Wang, Jiahang Cao, Lingfeng Zhang, Hao Cheng, Xiaoshuai Hao, Yiding Ji, Junwei Liang, Jian Tang, Renjing Xu, and Yijie Guo. Meshmimic: Geometry-aware humanoid motion learning through 3d scene reconstruction. *arXiv preprint arXiv:2602.15733*, 2026. 4

[46] Yuhong Zhang, Jing Lin, Ailing Zeng, Guanlin Wu, Shunlin Lu, Yurong Fu, Yuanhao Cai, Ruimao Zhang, Haoqian Wang, and Lei Zhang. Motion-x++: A large-scale multimodal 3d whole-body human motion dataset. *arXiv preprint arXiv:2501.05098*, 2025. 4

[47] Zhikai Zhang, Haofei Lu, Yunrui Lian, Ziqing Chen, Yun Liu, Chenghuai Lin, Han Xue, Zicheng Zeng, Zekun Qi, Shaolin Zheng, et al. Learning athletic humanoid tennis skills from imperfect human motion data. *arXiv preprint arXiv:2603.12686*, 2026. 4

[48] Ziyu Zhang, Sergey Bashkirov, Dun Yang, Yi Shi, Michael Taylor, and Xue Bin Peng. Physics-based motion imitation with adversarial differential discriminators. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, SA Conference Papers '25, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400721373. doi: 10.1145/3757377.3763819. URL https://doi.org/10.1145/3757377.3763819. 5

[49] Kaifeng Zhao, Gen Li, and Siyu Tang. Dartcontrol: A diffusion-based autoregressive motion model for real-time text-driven motion control. *arXiv preprint arXiv:2410.05260*, 2024. 5

[50] Shaoting Zhu, Baijun Ye, Jiaxuan Wang, Jiakang Chen, Ziwen Zhuang, Linzhan Mou, Runhan Huang, and Hang Zhao. Ttt-parkour: Rapid test-time training for perceptive robot parkour. *arXiv preprint arXiv:2602.02331*, 2026. 4

[51] Ziwen Zhuang, Shaoting Zhu, Mengjie Zhao, and Hang Zhao. Deep whole-body parkour. *arXiv preprint arXiv:2601.07701*, 2026. 4