GENTRACK: ONLINE CO-TRAINING OF TEXT-TO-MOTION GENERATION AND ZERO-SHOT HUMANOID TRACKING

Anonymous authors
Paper under double-blind review

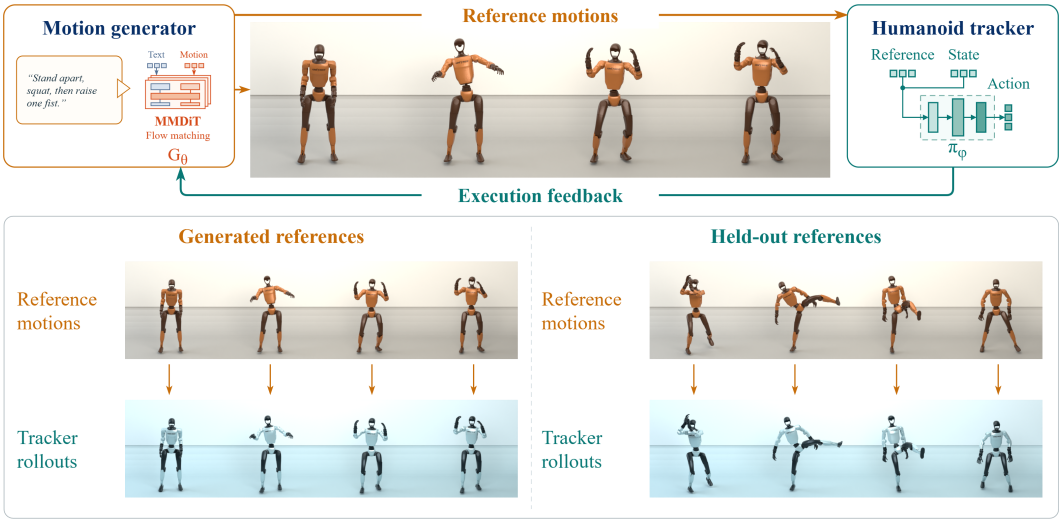

Figure 1: **Online generator–tracker co-training.** Generated robot-native references expand tracker supervision; their tracker rollouts provide execution feedback. At evaluation, the tracker executes held-out references.

ABSTRACT

Humanoid motion trackers need diverse reference motions for training, but collecting motion data and adapting it to a robot is costly. Pretrained text-to-motion generators can expand this training set by producing new robot motion references. However, motions learned from human data can remain difficult for robots to follow accurately, even after retargeting. These references can train the tracker, whose execution feedback can in turn improve the generator. We introduce GENTRACK, a unified online post-training framework that co-adapts robot-native text-to-motion generation and humanoid tracking. We first train a robot-native text-to-motion generator to initialize the motion source. GENTRACK then alternates two updates: generated motions augment public references for tracker training, and tracker rollouts provide execution rewards for refining the generator. A lagged tracker stays fixed within each generator phase and is refreshed between phases. Controlled comparisons with one-way updates and offline replay examine whether generation and tracking benefit from learning together. The results show that online co-training can improve tracking of unseen motions while making generated motions easier to execute. Replay comparisons further suggest that updating the motion source during training offers benefits beyond reusing a fixed generated dataset. These findings support online co-training as a way to expand motion supervision without collecting additional motion data while narrowing the gap between generated references and robot execution.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

A general-purpose humanoid controller should execute reference motions beyond its training set without motion-specific optimization or test-time adaptation. Recent generalist trackers have advanced zero-shot tracking on diverse motion collections (Yin et al., 2025; Han et al., 2025; Li et al., 2026c; Luo et al., 2025; Ma et al., 2026; Wang et al., 2026; Li et al., 2026d; Tao et al., 2026; Chen et al., 2026; Qi et al., 2026), yet their best results depend on embodied motion corpora of hundreds of millions to billions of frames (Luo et al., 2025; Qi et al., 2026). Scaling the breadth and quality of tracker supervision has thus become a central practical bottleneck for zero-shot humanoid tracking.

Expanding tracker coverage requires diverse motion references, yet collecting and adapting demonstrations remains costly. Direct robot demonstrations are limited by hardware rollout time and collection effort. Existing pipelines therefore acquire human motion from mocap, video, or teleoperation, retarget it to the robot, and train tracking policies in simulation (Fu et al., 2024; He et al., 2024a; 2025; NVLabs, 2025; Luo et al., 2025; Qi et al., 2026). This route provides large robot-space corpora, but extending their action, style, and composition coverage still requires additional data acquisition and processing.

Human-derived references also retain a *motion-to-execution gap*. Modern retargeters improve kinematic correspondence and downstream tracking (Araujo et al., 2025), but expressing a motion in robot coordinates does not ensure faithful execution. Contacts, joint coordination, and dynamic transitions can remain problematic under closed-loop control, motivating kinodynamic and physics-aware refinement (Chen et al., 2025; Zhao et al., 2026). Figure 2 illustrates this residual mismatch. Scaling tracker supervision thus requires both broader reference coverage and better compatibility between those references and robot execution.

Text-conditioned motion generators offer a reusable source of supervision. Pretrained models capture diverse actions, styles, and compositions (Guo et al., 2022; Tevet et al., 2023; Jiang et al., 2023; Zhang et al., 2023a; Chen et al., 2023; Guo et al., 2024), allowing additional references to be synthesized from language descriptions without collecting a demonstration for each variation. However, generators trained on human or retargeted data can inherit the motion-to-execution gap. Sampling from a fixed generator expands the reference pool but does not use execution experience to refine its output distribution. Generated supervision therefore needs both broad coverage and execution-grounded refinement.

One-way pipelines exploit each direction separately: generated motions can be executed by downstream physics policies (Rempe et al., 2026), frozen generative priors can supervise policy training (Zhang et al., 2025a), and fixed trackers can guide generator post-training (Yue et al., 2025; Zhang et al., 2026c). These approaches leave either the generator's distribution or the tracker's capabilities fixed. Joint generation–control training has also been explored for character animation, quadrupeds, and goal-conditioned humanoid skills (Xu et al., 2025; Gao et al., 2026; Debbad et al., 2026). We study the coupled problem of refining a broad text-conditioned robot-motion generator while extending a generalist tracker's coverage of unseen references.

The coupling arises because execution feedback depends on the tracker's current capabilities: a failed rollout may reflect either a problematic reference or a behavior the tracker has not yet mastered. Generated references expose the tracker to additional behaviors, while tracker rollouts reveal how those references execute on the robot. As the tracker learns, its feedback can reflect the changed capabilities; as the generator learns, it changes the references available for tracker training. This motivates online co-training, in which the reference distribution and tracking policy evolve together to expand supervision without additional demonstration collection while addressing the motion-to-execution gap.

We introduce GENTRACK, an online generator–tracker framework that closes this loop (Fig. 1). It alternates generator alignment with group-relative execution rewards and tracker updates on generated and public references. A lagged tracker scores candidates and remains fixed within each generator phase. Generated candidates passing integrity checks expand tracker supervision to motions at different levels of execution difficulty. Generator regularization, supervised rehearsal on existing paired data, and public references constrain drift during online co-training.

We validate GENTRACK on two publicly released humanoid trackers with distinct pretraining paradigms, ProtoMotions and SONIC, to test generalization across backbone designs. For each,

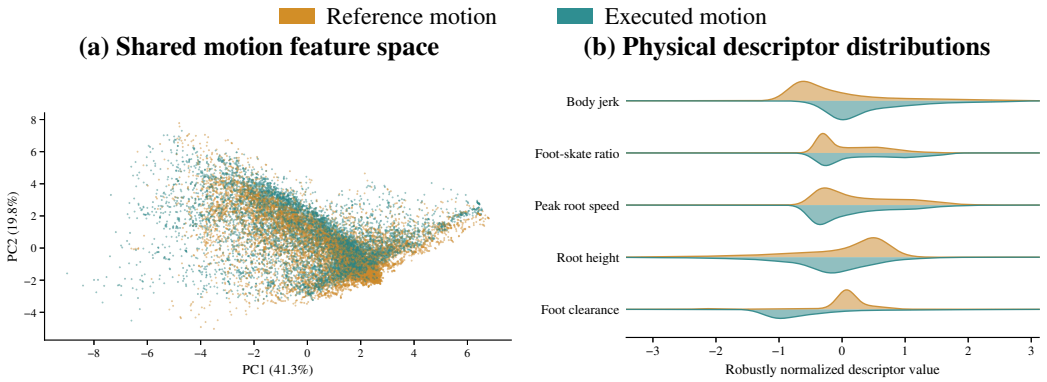

Figure 2: Motion-to-execution gap in diagnostic high-fidelity robot rollouts. **(a)** Shared PCA of 9,332 paired windows shows broad overlap with systematic local shifts. **(b)** Mirrored densities over all 10,369 paired windows reveal changes in body jerk, foot skating, root height, and foot clearance, with a smaller change in peak root speed. Additional separability diagnostics are reported in Appendix D.2.

we compare the pretrained checkpoint against reference-only continuation, offline replay, and one-way component updates to examine the effects of the two update directions. Evaluation uses three fixed zero-shot tracking splits (LAFAN1-G1, AMASS-test-G1, Wild-G1-clean) and held-out text–motion pairs from the test splits for generator evaluation; a common 30-FPS evaluator measures tracking success and trajectory errors, while official frozen-SONIC IsaacLab rollouts and TMR-G1 assess generator executability and semantic preservation. The SONIC branch improves mean generation metrics under frozen-SONIC evaluation and mean tracking performance relative to initialization. ProtoMotions improves mean Wild-G1-clean coverage and generator key-body accuracy, with trade-offs against offline replay and generator-only training. Together, these comparisons characterize how online post-training affects generated motion quality and held-out tracking across the two backbones.

Our contributions are:

- We introduce GENTRACK, a unified online post-training framework that couples text-conditioned, robot-native motion generation with zero-shot humanoid tracking to expand motion supervision without collecting additional demonstrations. By co-adapting the generator and tracker, GENTRACK connects reference generation with execution-grounded refinement.
- We develop an alternating training procedure that uses execution rewards to optimize the generator with FlowGRPO and mixes generated motions with public references to train the tracker. The scoring tracker is fixed within each generator phase, while generator regularization and rehearsal constrain distribution drift.
- We demonstrate that online co-training can improve both generated-motion executability and tracking generalization to unseen references. Experiments with SONIC and ProtoMotions evaluate generation under a fixed execution policy and tracking on three held-out motion splits, with one-way and offline-replay controls examining the benefits of the evolving training loop.

2 RELATED WORK

Humanoid motion tracking. Humanoid tracking converts motion references into physically executed behavior. Generalist trackers pursue coverage across varied references: Any2Track addresses cross-embodiment tracking, SONIC provides a whole-body tracking policy, and ProtoMotions supports physics-based motion imitation (Zhang et al., 2025b; Luo et al., 2025; NVLabs, 2025). Humanoid-GPT connects language conditioning with humanoid control (Qi et al., 2026), while BeyondMimic develops per-reference tracking specialists (Liao et al., 2025). These approaches mo-

tivate both broad motion coverage and accurate execution. GENTRACK complements tracker development by learning the distribution of generated references used for post-training alongside the tracking policy.

Execution-guided motion generation. Physics-aware generation reduces foot skating, penetration, unstable contacts, and dynamically infeasible poses. PhysDiff and CLoSD use simulation or controller loops, while RobotMDM, RLPF, Humanoid-R0, and PhysMoDPO optimize generators using a tracker-return surrogate, physical feedback, deployment-oriented rewards, or execution preferences (Yuan et al., 2023; Tevet et al., 2025; Serifi et al., 2024; Yue et al., 2025; Zhuang et al., 2025; Zhang et al., 2026c). Such feedback connects motion synthesis to execution, beyond the kinematic quality of generated references. GENTRACK uses execution feedback within an online loop that also updates the tracker receiving those references.

Joint generation and control. RoboForge and iterative closed-loop synthesis feed optimized or generated motions back into generator/control training (Yuan et al., 2026; Xu et al., 2026). Most closely, PARC alternates character-motion generation with physics-based tracking correction, QuadFM jointly trains a text-conditioned generator and controller for quadrupeds, and Humanoid-DART iteratively updates a goal-conditioned humanoid trajectory generator and tracker; a complementary humanoid locomotion system fine-tunes the tracker against a frozen generator (Xu et al., 2025; Gao et al., 2026; Debbad et al., 2026; Zhang et al., 2026d). Within this line of work, GENTRACK couples broad text-conditioned, robot-native generation with zero-shot humanoid tracking. Its evolving reference distribution expands tracker supervision without additional demonstration collection, while tracker feedback refines motion execution. We evaluate both generated-motion executability and tracker generalization to fixed, held-out references.

The supplementary material reviews text-to-motion generation, language-conditioned humanoids, retargeting, and tracker-only systems in greater detail.

3 METHOD

3.1 PROBLEM FORMULATION

Motion generation. Let $\mathcal{M}_R$ denote the reference-motion space of robot R. A generator with parameters θ models T-frame sequences $\mathbf{q} = (\mathbf{q}_1, \ldots, \mathbf{q}_T) \in \mathcal{M}_R$ given condition $\mathbf{y}$. We write $\mathbf{q} \sim p_\theta(\mathbf{q} \mid \mathbf{y})$, or equivalently $\mathbf{q} = G_\theta(\mathbf{y}, z)$ with stochastic input z. The condition may represent language or another motion specification; this work studies text-conditioned generation, where $\mathbf{y} = \mathbf{c}$. We instantiate G_θ as an MMDiT flow-matching model. In Fig. 3, small squares denote text and noisy-motion tokens; blue/orange rectangles denote their stream-specific transformations, the spanning bar denotes joint attention, and stacked outlines denote repeated Transformer blocks that predict the motion flow velocity.

Humanoid motion tracking. Given a reference $\mathbf{q}$, a tracker with parameters ϕ maps the robot state s_t and a reference command $g_t(\mathbf{q})$ to an action $a_t \sim \pi_\phi(\cdot \mid s_t, g_t)$ at time step t. The rollout operator $\mathcal{E}$ produces the closed-loop trajectory $\boldsymbol{\tau} = \mathcal{E}(\pi_\phi, \mathbf{q})$. Tracking maximizes

$$J_{\text{track}}(\phi; \mathbf{q}) = \mathbb{E}\left[\sum_{t=1}^{T} \gamma^{t-1} r_{\text{track}}(s_t, \mathbf{q}_t)\right]. \tag{1}$$

where $\gamma \in [0, 1]$ is the discount factor, r_{track} is the native per-step tracking reward, and the expectation is over rollouts. In Fig. 3, squares denote inputs/actions and bars denote learned policy transformations within the dashed network outline. Merged lines indicate conditioning on reference and state; each tracker retains its native architecture. Here, zero-shot tracking evaluates a fixed policy on references held out from GenTrack post-training, without motion-specific optimization or test-time adaptation. This definition does not certify exclusion from every released checkpoint's pretraining data.

Robot-native parameterization. Each frame uses a deterministic 38-dimensional G1 parameterization: a three-dimensional root channel, continuous 6D pelvis rotation, and 29 actuated joints.

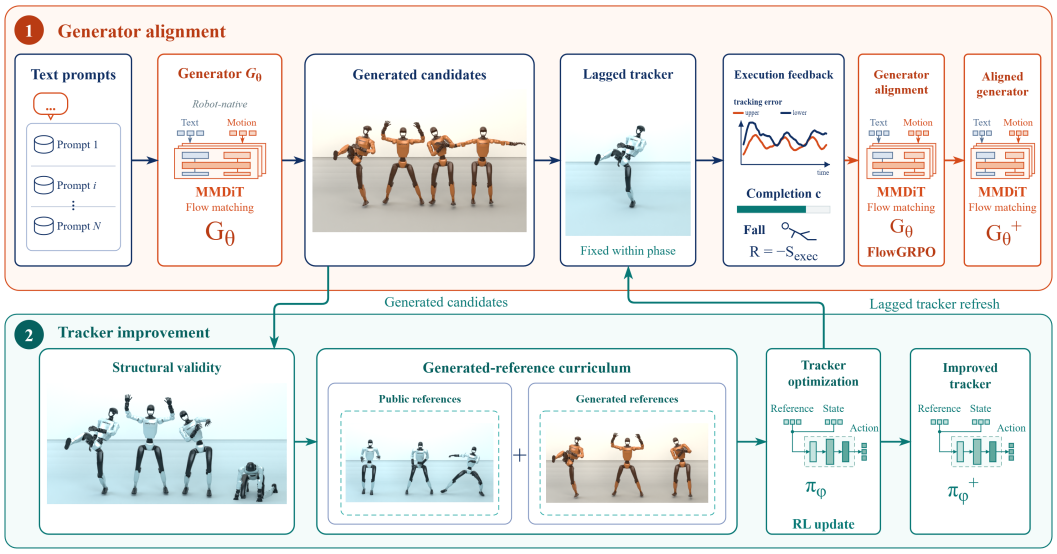

Figure 3: **GENTRACK pipeline.** Generated references extend the tracker curriculum; execution feedback aligns the generator through FlowGRPO with a lagged tracker frozen within each phase. Curves show smoothed upper/lower-body position errors from one rollout; icons denote completion and falls. Model icons summarize MMDiT text–motion processing and the tracker input–action interface. Superscript + marks an updated model; the central + mixes public and generated references.

Per-clip planar position and heading are canonicalized; planar motion is stored as displacement while height remains absolute. One-time offline GMD retargeting supplies paired initialization data, but no retargeter runs inside the online loop. Supplementary Section B gives preprocessing details.

3.2 FRAMEWORK OVERVIEW

We fine-tune $G_0 = G_{\theta_0}$ on paired text–motion data so varied prompts yield varied, relevant references (Table 3). An existing robot-native generator with suitable text following can also initialize the framework. Starting with pretrained tracker $\pi_0 = \pi_{\phi_0}$, GENTRACK alternates generator alignment and tracker improvement (Fig. 3). Each round samples and executes *generated candidates*; those passing integrity checks enter tracker training as *generated references*.

Tracker updates mix public and accumulated generated references. The lagged tracker scores candidates using completion, joint/root errors, and falls, remaining frozen within each generator phase. Frozen-generator KL regularization and supervised rehearsal with G_0 distillation limit drift. The alternating updates evolve both the generated pool and its execution reward.

3.3 GENERATED-REFERENCE CURRICULUM FOR TRACKER OPTIMIZATION

Generated references extend an existing tracker without requiring a new large tracker-specific reference corpus. Let $\mathcal{D}_{\text{base}}$ be the fixed public-reference pool and $\mathcal{P}_{\text{train}} = \{\mathbf{c}_i\}_{i=1}^{N}$ the pool of N training prompts. For each sampled prompt $\mathbf{c}_i$, we draw K noise inputs $z_{i,k}$, $k = 1, \ldots, K$, and generate $G_\theta(\mathbf{c}_i, z_{i,k})$. These candidates form $\mathcal{D}_{\text{gen}}$; the tracker trains on

$$\mathcal{D}_{\text{track}} = \mathcal{D}_{\text{base}} \cup \mathcal{D}_{\text{gen-valid}}, \tag{2}$$

where $\mathcal{D}_{\text{gen-valid}}$ contains generated candidates that pass reference admission, which checks integrity (the Structural validity block in Fig. 3). Offline curation of the generator's training corpus addresses scene compatibility and temporal quality (Supplementary Section B). These references provide practice across levels of execution difficulty. Tracker updates sample from both $\mathcal{D}_{\text{base}}$ and the accumulated $\mathcal{D}_{\text{gen-valid}}$ using the backbone's native motion sampler. Execution feedback progressively refines this evolving reference curriculum.

3.4 Execution Reward for Generator Alignment

Tracker rollouts turn robot execution into a learning signal for text-to-motion generation. At round r, we use a delayed checkpoint as the lagged tracker $\bar{\pi}^{(r)}$, freeze it throughout generator optimization, and define

$$R(\mathbf{q}; \bar{\pi}^{(r)}) = -S_{\text{exec}}(\mathbf{q}; \bar{\pi}^{(r)}). \tag{3}$$

For completion fraction c, maximum joint error e_{joint}, mean root-trajectory error e_{traj}, and root-displacement error e_{disp},

$$S_{\text{exec}} = (1 - c) + [e_{\text{joint}}]_2 + [e_{\text{traj}}/0.5]_2 + 0.5[e_{\text{disp}}/0.5]_2 + 2\mathbb{I}_{\text{fall}}, \tag{4}$$

where $[x]_2 = \min(x, 2)$ clips from above and $\mathbb{I}_{\text{fall}}$ is 1 if the scoring rollout registers a fall and 0 otherwise. The current tracker has zero reward weight, and velocity, amplitude, and binary-success gates are not used by the main protocol. Completion and fall terms penalize interrupted or unstable execution; joint and root errors distinguish faithful tracking among completed rollouts. Clipping bounds each error contribution. Success-only and retention ablations examine how execution feedback and distribution preservation affect the results.

3.5 Generator Optimization

Given K samples from the same prompt, GenTrack updates the generator with a group-relative FlowGRPO objective. Rewards are normalized within each prompt group, and sampled trajectories are replayed for multiple clipped policy-ratio updates. A frozen-generator KL penalty and supervised flow-matching rehearsal with G_0 distillation on the original train-only pairs constrain drift from the pretrained motion distribution (Yue et al., 2025; Liu et al., 2025a). TMR-G1 retrieval and diversity are held-out metrics and do not enter the reward. The supplement reports retention weights, replay updates, and ablations assessing semantic and distribution preservation.

3.6 Training Schedule

We alternate generator and tracker updates for 12 rounds. Each round (1) samples K same-prompt candidates and checks reference integrity; (2) scores valid candidates with the lagged tracker and applies FlowGRPO with retention regularization; (3) accumulates generated references and updates the tracker on the mixed reference pool using its native objective; and (4) exports that tracker for the next round. We log reward variance, policy-ratio movement, retention updates, exported references, and validation performance at every boundary. Phase-wise freezing fixes the lagged tracker within an update phase; the objective still changes across rounds.

4 Experiments

We evaluate whether online generated references extend held-out tracker coverage and whether execution feedback improves generated-motion executability under a frozen executor. One-way and offline controls examine the two update directions.

4.1 Experimental Setup

Protocol and data. All experiments use simulated Unitree G1 and 30-FPS exports. We instantiate GenTrack from fixed ProtoMotions (NVLabs, 2025) and SONIC (Luo et al., 2025) checkpoints; within each backbone, trainable rows share initialization, simulator, optimizer, transition budget, validation, and test manifests. Tracker post-training uses 13,337 public AMASS/LAFAN1 references and generated motions. The 357,472 GMD-retargeted G1 pairs from an internal human-motion corpus serve only generator initialization and rehearsal. Tracking evaluation uses the fixed LAFAN1-G1, AMASS-test-G1, and Wild-G1-clean splits. Generator evaluation uses 1,024 held-out text–motion pairs from the test splits; these pairs are excluded from generator, tracker post-training, and TMR-G1 training data. External rollouts receive the same post-hoc processing. We evaluate all generators with the same frozen, released SONIC policy, chosen for its strong tracking capability across diverse motions without motion-specific retraining. Using a capable generalist executor helps assess generated-motion quality while reducing the influence of limited tracker coverage. Offline controls also match accepted-motion counts or tracker transitions as applicable; released native

Table 1: Zero-shot G1 tracking under a common fall-only protocol; errors average motions equally, including failures. Primary trainable rows share initialization and budget within each backbone; G_{final} replay is an additional offline control. Bold marks the best primary trainable mean. LAFAN1-G1, AMASS-test-G1, and Wild-G1-clean contain 40, 138, and 1,024 motions. Protocol: Supplementary Section C. Superscripts give ±95% CI half-widths across five runs.

Method	Fall-only SR (%) ↑			All-trajectory error ↓			
	LAFAN1	AMASS-test	Wild-G1	MPJPE (mm)	E_g (mm)	MPJVE (m/s)	RootVelErr (m/s)
Baselines							
Any2Track (Zhang et al., 2025b)	100.0	5.1	10.4	320.9	1309.1	0.720	0.632
BeyondMimic (Liao et al., 2025)	87.5	93.0	61.0	99.4	347.6	0.251	0.279
Humanoid-GPT (Qi et al., 2026)	85.0	83.3	71.4	128.1	1134.7	0.689	0.644
ProtoMotions (π_0) (NVLabs, 2025)	75.0	81.2	45.9	142.2	789.8	0.320	0.466
SONIC (Luo et al., 2025)	85.0	79.0	47.2	126.2	814.2	0.308	0.423
ProtoMotions post-training							
Ref. only	$75.0^{\pm 3.1}$	$80.4^{\pm 1.3}$	$45.7^{\pm 1.0}$	$141.9^{\pm 2.6}$	$786.5^{\pm 10.0}$	$0.321^{\pm 0.006}$	$0.466^{\pm 0.009}$
G_0 replay	$\mathbf{77.5}^{\pm 2.2}$	$79.0^{\pm 1.4}$	$46.8^{\pm 0.9}$	$140.0^{\pm 2.5}$	$\mathbf{772.4}^{\pm 14.0}$	$\mathbf{0.319}^{\pm 0.006}$	$\mathbf{0.461}^{\pm 0.015}$
G_{final} replay	$77.5^{\pm 2.2}$	$78.3^{\pm 1.7}$	$46.4^{\pm 0.9}$	$140.7^{\pm 2.4}$	$756.2^{\pm 13.8}$	$0.318^{\pm 0.007}$	$0.456^{\pm 0.012}$
GENTRACK	$75.0^{\pm 3.1}$	$\mathbf{81.2}^{\pm 1.8}$	$\mathbf{47.3}^{\pm 1.2}$	$\mathbf{139.3}^{\pm 2.8}$	$775.4^{\pm 16.8}$	$0.320^{\pm 0.007}$	$0.466^{\pm 0.016}$
SONIC post-training							
Ref. only	$82.5^{\pm 2.2}$	$\mathbf{79.7}^{\pm 0.9}$	$45.9^{\pm 1.0}$	$131.8^{\pm 2.3}$	$867.6^{\pm 9.5}$	$0.314^{\pm 0.006}$	$0.436^{\pm 0.008}$
G_0 replay	$85.0^{\pm 3.1}$	$78.3^{\pm 1.3}$	$45.5^{\pm 0.9}$	$133.3^{\pm 2.9}$	$847.6^{\pm 14.5}$	$0.321^{\pm 0.008}$	$0.433^{\pm 0.011}$
G_{final} replay	$87.5^{\pm 3.1}$	$76.1^{\pm 1.7}$	$47.8^{\pm 0.8}$	$126.2^{\pm 2.5}$	$841.0^{\pm 12.5}$	$0.314^{\pm 0.006}$	$0.434^{\pm 0.009}$
GENTRACK	$\mathbf{90.0}^{\pm 3.8}$	$\mathbf{79.7}^{\pm 1.4}$	$\mathbf{48.0}^{\pm 1.4}$	$\mathbf{124.1}^{\pm 2.9}$	$\mathbf{807.2}^{\pm 19.2}$	$\mathbf{0.308}^{\pm 0.007}$	$\mathbf{0.423}^{\pm 0.009}$

success flags are excluded from the common comparison. We report five-run means and 95% confidence intervals for trained components, using one shared G_0.

Metrics. We evaluate trackers by common physical-coverage and motion-fidelity measures, and generators by frozen-SONIC executability plus TMR-G1 semantic and distribution measures. Full definitions, body sets, alignment, thresholds, aggregation, and diagnostic-only rules are given in supplementary Section C.1 and Table 4.

4.2 Main Tracker Post-Training Results

Table 1 compares matched post-training variants with external trackers; BeyondMimic is a per-reference specialist. With ProtoMotions, frozen-G_0 replay trades AMASS-test SR for LAFAN1/Wild-G1-clean gains; GENTRACK matches initialization on AMASS-test, attains the highest mean Wild-G1-clean SR among trained variants, and lowers mean MPJPE from 142.2 to 139.3 mm, with higher mean E_g than frozen-G_0 replay. With SONIC, mean SR rises from 85.0/79.0/47.2 to 90.0/79.7/48.0, MPJPE falls from 126.2 to 124.1 mm, and E_g from 814.2 to 807.2 mm; both velocity errors match initialization. Errors include failed trajectories on fixed references, whose content the generator cannot simplify. Against frozen-G_0 replay, the clearest differences are in MPJPE (133.3 to 124.1 mm) and E_g (847.6 to 807.2 mm), with separated 95% intervals. Against G_{final} replay, every reported mean favors the online SONIC row, although the Wild SR gap is 0.2 percentage points with overlapping intervals. These means suggest benefits from updating references during training beyond G_{final} replay. Supplementary Table 5 reports per-split errors.

4.3 Main Generator Results

Table 2 compares generators using identical prompts, sampling noise, and motion lengths, evaluated by the same frozen SONIC policy. Tracker-filtered SFT selects its training candidates with frozen ProtoMotions π_0. GENTRACK (SONIC) raises mean execution success from 92.58% to 94.43% and lowers joint error from 0.159 to 0.152 rad. The largest relative accuracy gain is in key-body error: 0.410 to 0.325 m, a 20.7% reduction. Mean R@1 rises from 0.774 to 0.783; R@2 and FID change modestly from 0.888/0.023 to 0.890/0.020. Diversity remains close to initialization (36.371 versus

Table 2: Generator post-training on 1,024 matched cases: (a) executability under frozen SONIC and (b) text–motion quality under TMR-G1. Bold marks best means, including ties; Diversity is descriptive. Superscripts give ±95% CI half-widths across five runs. Metric definitions: Appendix C.1.

(a) Executability

Generator	Succ. (%) ↑	E_{joint} (rad) ↓	E_{key} (m) ↓
G_0	92.58	0.159	0.410
Tracker-filtered SFT	**96.97**$^{\pm 0.52}$	**0.149**$^{\pm 0.004}$	0.348$^{\pm 0.007}$
Frozen-SONIC FlowGRPO	90.92$^{\pm 0.54}$	0.158$^{\pm 0.003}$	0.363$^{\pm 0.008}$
GENTRACK (ProtoMotions)	93.55$^{\pm 0.81}$	0.160$^{\pm 0.004}$	0.399$^{\pm 0.009}$
GENTRACK (SONIC)	94.43$^{\pm 0.78}$	0.152$^{\pm 0.003}$	**0.325**$^{\pm 0.010}$

(b) Text–motion quality

Generator	R@1 ↑	R@2 ↑	R@3 ↑	MM-Dist ↓	FID ↓	Diversity
G_0	0.774	0.888	**0.931**	**18.293**	0.023	36.375
Tracker-filtered SFT	0.771$^{\pm 0.003}$	0.886$^{\pm 0.002}$	0.929$^{\pm 0.002}$	18.450$^{\pm 0.067}$	0.028$^{\pm 0.002}$	36.264$^{\pm 0.139}$
Frozen-SONIC FlowGRPO	0.767$^{\pm 0.004}$	0.875$^{\pm 0.004}$	0.920$^{\pm 0.002}$	18.612$^{\pm 0.096}$	0.027$^{\pm 0.004}$	36.433$^{\pm 0.177}$
GENTRACK (ProtoMotions)	0.782$^{\pm 0.005}$	0.889$^{\pm 0.004}$	0.930$^{\pm 0.003}$	18.299$^{\pm 0.059}$	**0.020**$^{\pm 0.003}$	36.373$^{\pm 0.261}$
GENTRACK (SONIC)	**0.783**$^{\pm 0.005}$	**0.890**$^{\pm 0.003}$	0.930$^{\pm 0.004}$	18.302$^{\pm 0.085}$	**0.020**$^{\pm 0.003}$	36.371$^{\pm 0.277}$

36.375), as do R@3 and MM-Dist (0.930 and 18.302). Execution success and accuracy improve alongside retained text–motion alignment and diversity.

Tracker-filtered SFT attains higher mean success (96.97%) and lower mean joint error (0.149 rad). GENTRACK (SONIC) instead has lower key-body error (0.325 versus 0.348 m), higher R@1, and lower FID (0.020 versus 0.028), with separated intervals for these three metrics. Its mean R@2 and R@3 are also higher, with overlapping intervals. Success rate alone therefore misses differences in motion fidelity and semantic quality. ProtoMotions shows favorable mean changes in success, key-body error, R@1/2, and FID over G_0, with a slightly higher joint error.

4.4 ABLATION STUDY

Supplementary Table 6 separates update directions within ProtoMotions. Generator-only training reaches 93.95 success, $E_{\text{key}} = 0.381$, and R@1/FID of 0.783/0.020. Its means favor generation over the full loop, which instead raises tracker Wild-G1-clean SR to 47.3. The key-body error difference is clearer than the smaller success and retrieval gaps. This branch exhibits trade-offs between the two components. The SONIC branch has favorable mean changes over frozen-SONIC FlowGRPO in Table 2; cross-backbone comparisons do not isolate mutual adaptation. Among matched objective controls, dense feedback has higher mean generator success than success-only feedback (93.55 versus 93.12) and lower key-body error (0.399 versus 0.406 m), with overlapping intervals. Reward-weighted SFT favors mean execution metrics at the expense of retrieval and FID. The full loop has more favorable mean success, R@1, and FID than Diffusion-DPO, with overlapping intervals. Retention ablations (Table 7) show clearer separation in content metrics: removing both KL and rehearsal lowers retrieval and diversity and increases FID, despite higher mean execution success.

4.5 QUALITATIVE ANALYSIS

Figure 4 compares matched generator prompts, noise, and lengths, and a fixed held-out tracking reference. Generator refinement removes artifacts such as abrupt joint reversals; tracker refinement improves execution through difficult crouches, turns, and contact transitions. These selected cases illustrate Tables 1 and 2, without establishing their frequency or cause. Appendix D.2 gives further analysis.

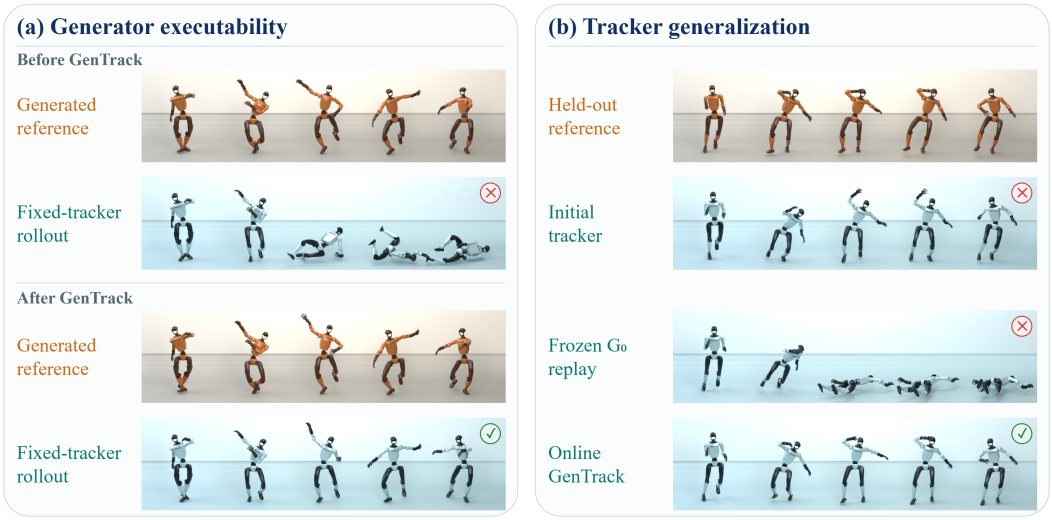

Figure 4: **Qualitative effects of online co-training. (a) Generator executability.** With matched prompts and sampling seeds, frozen SONIC fails on the initial reference but executes the aligned generator's reference. **(b) Tracker generalization.** On the same held-out Wild-G1-clean reference, the initial and frozen-G_0 replay trackers fail; online GENTRACK succeeds. Red circled crosses mark failures; green circled check marks indicate success. Rows show five synchronized, uniformly spaced phases. Snapshots are individually pelvis-centered and yaw-normalized for display; Table 1 uses the metric-specific trajectory alignments.

4.6 MOTION-TO-EXECUTION GAP AND LIMITATIONS

Aggregate tracking metrics do not reveal *how* retargeted references and executed motions differ. Figure 2 characterizes this residual gap (which motivates GENTRACK) through shared PCA (first two components explain 61.1% of pooled variance) and a motion-grouped classifier (AUC 0.996 ± 0.004), confirming the gap is systematic yet the domains share substantial structure. Per-descriptor densities localize the mismatch to jerk, skating, root height, and foot clearance. These descriptive diagnostics motivate evaluating execution beyond binary success; Supplementary Section D.2 provides details.

Limitations. Our experiments focus on simulated Unitree G1 with ProtoMotions and SONIC tracking backbones. Generator comparisons use a common frozen SONIC executor; evaluation with additional execution policies would broaden this assessment. Extending the framework to other robot morphologies requires adapting the motion representation and tracking interface. Physical deployment remains future work because hardware resources are currently unavailable. A practical next step is to evaluate the learned policies under real-world sensing, actuation, and contact conditions.

5 CONCLUSION

We presented GENTRACK to expand motion supervision for humanoid tracking while narrowing the gap between generated references and robot execution. Generated motions broaden tracker training, while execution feedback refines the generator. Controlled experiments show that this interaction can improve both tracking of unseen motions and generated-motion executability. Replay comparisons suggest value in an evolving source of motion supervision. These findings support treating text-conditioned motion generation and humanoid tracking as coupled learning tasks: the generator supplies additional supervision without new motion collection, and the tracker helps refine that supervision through execution experience.

AI USAGE DISCLOSURE

During the development of GENTRACK, we used AI tools to polish the writing of this paper and to assist with coding for the experiments. The authors take responsibility for the content of the paper and the experimental code.

REFERENCES

Joao Pedro Araujo, Yanjie Ze, Pei Xu, Jiajun Wu, and C. Karen Liu. Retargeting matters: General motion retargeting for humanoid motion tracking. *arXiv preprint arXiv:2510.02252*, 2025.

Jiacheng Bao, Haoran Yang, Yucheng Xin, Junhong Liu, Yuecheng Xu, Han Liang, Pengfei Han, Xiaoguang Ma, Dong Wang, and Bin Zhao. PhyGile: Physics-prefix guided motion generation for agile general humanoid motion tracking. *arXiv preprint arXiv:2603.19305*, 2026.

German Barquero, Sergio Escalera, and Cristina Palmero. Seamless human motion composition with blended positional encodings. In *CVPR*, pp. 457–469, 2024.

Jianuo Cao, Yuxin Chen, and Masayoshi Tomizuka. CLAW: Composable language-annotated whole-body motion generation. *arXiv preprint arXiv:2604.11251*, 2026.

Maiyue Chen, Kaihui Wang, Bo Zhang, Xihan Ma, Zhiyuan Yang, Yi Ren, Qijun Huang, Zihao Zhu, Yucheng Wang, and Zhizhong Su. HoloMotion-1 technical report. *arXiv preprint arXiv:2605.15336*, 2026.

Wenshuo Chen, Hongru Xiao, Erhang Zhang, Lijie Hu, Lei Wang, Mengyuan Liu, and Chen Chen. SATO: Stable text-to-motion framework. *arXiv preprint arXiv:2405.01461*, 2024.

Xin Chen, Biao Jiang, Wen Liu, Zilong Huang, Bin Fu, Tao Chen, and Gang Yu. Executing your commands via motion diffusion in latent space. In *CVPR*, pp. 18000–18010, 2023.

Xingyu Chen, Hanyu Wu, Sikai Wu, Mingliang Zhou, Diyun Xiang, Haodong Zhang, Yangchen Zhou, Yukang Gao, Yi Gu, and Renjing Xu. A scalable whole-body motion transfer via implicit kinodynamic motion retargeting. *arXiv preprint arXiv:2509.15443*, 2025.

Hanbyel Cho, Sang-Hun Kim, Jeonguk Kang, and Donghan Koo. SafeFlow: Real-time text-driven humanoid whole-body control via physics-guided rectified flow and selective safety gating. *arXiv preprint arXiv:2603.23983*, 2026.

Wenxun Dai, Ling-Hao Chen, Jingbo Wang, Jinpeng Liu, Bo Dai, and Yansong Tang. MotionLCM: Real-time controllable motion generation via latent consistency model. In *ECCV*, 2024.

Pranav Debbad, Kanish Thiagarajan, Victor Dhédin, Shafeef Omar, and Majid Khadiv. Humanoid-DART: Humanoid loco-manipulation using diffusion-guided augmentation through relabeling and tracking. *arXiv preprint arXiv:2606.26855*, 2026.

Zipeng Fu, Qingqing Zhao, Qi Wu, Gordon Wetzstein, and Chelsea Finn. HumanPlus: Humanoid shadowing and imitation from humans. *arXiv preprint arXiv:2406.10454*, 2024.

Li Gao, Fuzhi Yang, Jianhui Chen, Liu Liu, Yao Zheng, Yang Cai, and Ziqiao Li. QuadFM: Foundational text-driven quadruped motion dataset for generation and control. *arXiv preprint arXiv:2603.24021*, 2026.

Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3d human motions from text. In *CVPR*, 2022.

Chuan Guo, Yuxuan Mu, Muhammad Gohar Javed, Sen Wang, and Li Cheng. MoMask: Generative masked modeling of 3d human motions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1900–1910, 2024.

Jinrui Han, Weiji Xie, Jiakun Zheng, Jiyuan Shi, Weinan Zhang, Ting Xiao, and Chenjia Bai. KungfuBot2: Learning versatile motion skills for humanoid whole-body control. *arXiv preprint arXiv:2509.16638*, 2025.

Sudarshan Harithas, Sangkyung Kwak, Pushkal Katara, Srujan Deolasee, Dvij Kalaria, Srinath Sridhar, Sai Vemprala, Ashish Kapoor, and Jonathan Chung-Kuan Huang. DreamControl-v2: Simpler and scalable autonomous humanoid skills via trainable guided diffusion priors. *arXiv preprint arXiv:2604.00202*, 2026.

Felix G. Harvey, Mike Yurick, Derek Nowrouzezahrai, and Christopher Pal. Robust motion inbetweening. *ACM Transactions on Graphics*, 2020.

Tairan He, Zhengyi Luo, Xialin He, Wenli Xiao, Chong Zhang, Weinan Zhang, Kris Kitani, Changliu Liu, and Guanya Shi. OmniH2O: Universal and dexterous human-to-humanoid whole-body teleoperation and learning. *arXiv preprint arXiv:2406.08858*, 2024a.

Tairan He, Zhengyi Luo, Wenli Xiao, Chong Zhang, Kris Kitani, Changliu Liu, and Guanya Shi. Learning human-to-humanoid real-time whole-body teleoperation. *arXiv preprint arXiv:2403.04436*, 2024b.

Tairan He, Jiawei Gao, Wenli Xiao, Yuanhang Zhang, Zi Wang, Jiashun Wang, Zhengyi Luo, Guanqi He, Nikhil Sobanbabu, Chaoyi Pan, Zeji Yi, Guannan Qu, Kris Kitani, Jessica Hodgins, Linxi Fan, Yuke Zhu, Changliu Liu, and Guanya Shi. ASAP: Aligning simulation and real-world physics for learning agile humanoid whole-body skills. *arXiv preprint arXiv:2502.01143*, 2025.

Tianchen Huang, Feiyang Yuan, Junchi Gu, Shurui Fang, Xiaohu Zhang, Yu Wang, Wei Gao, and Shiwu Zhang. Human2Humanoid: Physics-aware cross-morphology motion retargeting for humanoid robots. *arXiv preprint arXiv:2606.03476*, 2026.

Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. MotionGPT: Human motion as a foreign language. In *NeurIPS*, 2023.

Bin Li, Ruichi Zhang, Han Liang, Jingyan Zhang, Juze Zhang, Xin Chen, and Jingya Wang. MIND: Multi-scale intent diffusion for text-driven physics-based humanoid control. *arXiv preprint arXiv:2605.26006*, 2026a.

Peng Li, Zihan Zhuang, Yangfan Gao, Yi Dong, Sixian Li, Changhao Jiang, Shihan Dou, Zhiheng Xi, Enyu Zhou, Jixuan Huang, Hui Li, Jingjing Gong, Xingjun Ma, Tao Gui, Zuxuan Wu, Qi Zhang, Xuanjing Huang, Yu-Gang Jiang, and Xipeng Qiu. FRoM-W1: Towards general humanoid whole-body control with language instructions. *arXiv preprint arXiv:2601.12799*, 2026b.

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 *International Conference on Learning Representations*, 2026c.

Yuhan Li, Peiyuan Zhi, Yunshen Wang, Tengyu Liu, Sixu Yan, Wenyu Liu, Xinggang Wang, Baoxiong Jia, and Siyuan Huang. OmniTrack: General motion tracking via physics-consistent reference. *arXiv preprint arXiv:2602.23832*, 2026d.

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. *ICLR*, 2026e.

Qiayuan Liao, Takara E. Truong, Xiaoyu Huang, Yuman Gao, 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.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-GRPO: Training flow matching models via online rl. *arXiv preprint arXiv:2505.05470*, 2025a.

Zhirui Liu, Kaiyang Ji, Ke Yang, Yahao Fan, Jingyi Yu, Ye Shi, and Jingya Wang. Commanding humanoid by free-form language: A large language action model with unified motion vocabulary. *arXiv preprint arXiv:2511.22963*, 2025b.

Shunlin Lu, Ling-Hao Chen, Ailing Zeng, Jing Lin, Ruimao Zhang, Lei Zhang, and Heung-Yeung Shum. HumanTOMATO: Text-aligned whole-body motion generation. In *ICML*, 2024.

Zhengyi Luo, Ye Yuan, Tingwu Wang, Chenran Li, Fernando Castañeda, Sirui Chen, Zi-Ang Cao, Jiefeng Li, David Minor, Qingwei Ben, Jinhyung Park, David Sami, Zi Wang, Xingye Da, Runyu Ding, Cyrus Hogg, Lina Song, Edy Lim, Eugene Jeong, Tairan He, Haoru Xue, Wenli Xiao, Simon Yuen, Jan Kautz, Yan Chang, Umar Iqbal, Linxi Fan, and Yuke Zhu. SONIC: Supersizing motion tracking for natural humanoid whole-body control. *arXiv preprint arXiv:2511.07820*, 2025.

Yubiao Ma, Han Yu, Jiayin Xie, Changtai Lv, Qiang Luo, Chi Zhang, Yunpeng Yin, Boyang Xing, Xuemei Ren, and Dongdong Zheng. Robust and generalized humanoid motion tracking. *arXiv preprint arXiv:2601.23080*, 2026.

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*, 2019.

NVLabs. ProtoMotions: Gpu-accelerated simulation and learning for humanoids. `https://github.com/NVlabs/ProtoMotions`, 2025.

Mathis Petrovich, Michael J. Black, and Gül Varol. Tmr: Text-to-motion retrieval using contrastive 3d human motion synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

Ekkasit Pinyoanuntapong, Pu Wang, Minwoo Lee, and Chen Chen. MMM: Generative masked motion model. In *CVPR*, 2024.

Matthias Plappert, Christian Mandery, and Tamim Asfour. The KIT motion-language dataset. In *Big Data*, 2016.

Zekun Qi, Xuchuan Chen, Dairu Liu, Chenghuai Lin, Yunrui Lian, Sikai Liang, Zhikai Zhang, Yu Guan, Jilong Wang, Wenyao Zhang, Xinqiang Yu, He Wang, and Li Yi. Humanoid-GPT: Scaling data and structure for zero-shot motion tracking. *arXiv preprint arXiv:2606.03985*, 2026.

Davis Rempe, Mathis Petrovich, Ye Yuan, Haotian Zhang, Xue Bin Peng, Yifeng Jiang, Tingwu Wang, Umar Iqbal, David Minor, Michael de Ruyter, Jiefeng Li, Chen Tessler, Edy Lim, Eugene Jeong, Sam Wu, Ehsan Hassani, Michael Huang, Jin-Bey Yu, Chaeyeon Chung, Lina Song, Olivier Dionne, Jan Kautz, Simon Yuen, and Sanja Fidler. Kimodo: Scaling controllable human motion generation. *arXiv preprint arXiv:2603.15546*, 2026.

Agon Serifi, Ruben Grandia, Espen Knoop, Markus Gross, and Moritz Bächer. Robot Motion Diffusion Model: Motion generation for robotic characters. In *SIGGRAPH Asia 2024 Conference Papers*, pp. 1–9, 2024. doi: 10.1145/3680528.3687626.

Zelin Tao, Zeran Su, Peiran Liu, Jingkai Sun, Wenqiang Que, Jiahao Ma, Jialin Yu, Jiahang Cao, Pihai Sun, Hao Liang, Gang Han, Wen Zhao, Zhiyuan Xu, Jian Tang, Qiang Zhang, and Yijie Guo. Heracles: Bridging precise tracking and generative synthesis for general humanoid control. *arXiv preprint arXiv:2603.27756*, 2026.

Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H. Bermano. Human motion diffusion model. In *ICLR*, 2023.

Guy Tevet, Sigal Raab, Setareh Cohan, Daniele Reda, Zhengyi Luo, Xue Bin Peng, Amit H. Bermano, and Michiel van de Panne. CLoSD: Closing the loop between simulation and diffusion for multi-task character control. In *ICLR*, 2025.

Zepeng Wang, Jiangxing Wang, Shiqing Yao, Yu Zhang, Ziluo Ding, Ming Yang, Yuxuan Wang, Haobin Jiang, Chao Ma, Xiaochuan Shi, and Zongqing Lu. General humanoid whole-body control via pretraining and fast adaptation. *arXiv preprint arXiv:2602.11929*, 2026.

Weiji Xie, Jiakun Zheng, Jinrui Han, Jiyuan Shi, Weinan Zhang, Chenjia Bai, and Xuelong Li. TextOp: Real-time interactive text-driven humanoid robot motion generation and control. *arXiv preprint arXiv:2602.07439*, 2026.

Michael Xu, Yi Shi, KangKang Yin, and Xue Bin Peng. PARC: Physics-based augmentation with reinforcement learning for character controllers. In *SIGGRAPH Conference Papers*, 2025. doi: 10.1145/3721238.3730616.

Weisheng Xu, Qiwei Wu, Jiaxi Zhang, Jing Tan, Yangfan Li, Yuetong Fang, Jiaqi Xiong, Kai Wu, Rong Ou, and Renjing Xu. Iterative closed-loop motion synthesis for scaling the capabilities of humanoid control. In *CVPR*, pp. 16398–16407, 2026.

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.

Xichen Yuan, Zhe Li, Bofan Lyu, Kuangji Zuo, Yanshuo Lu, Gen Li, and Jianfei Yang. RoboForge: Physically optimized text-guided whole-body locomotion for humanoids. *arXiv preprint arXiv:2603.17927*, 2026.

Ye Yuan, Jiaming Song, Umar Iqbal, Arash Vahdat, and Jan Kautz. PhysDiff: Physics-guided human motion diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16010–16021, 2023.

Junpeng Yue, Zepeng Wang, Yuxuan Wang, Weishuai Zeng, Jiangxing Wang, Xinrun Xu, Yu Zhang, Sipeng Zheng, Ziluo Ding, and Zongqing Lu. Rl from physical feedback: Aligning large motion models with humanoid control. *arXiv preprint arXiv:2506.12769*, 2025.

Haodong Zhang, Liang Zhang, Zhenghan Chen, Lu Chen, Yue Wang, and Rong Xiong. Natural humanoid robot locomotion with generative motion prior. *arXiv preprint arXiv:2503.09015*, 2025a.

Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Yong Zhang, Hongwei Zhao, Hongtao Lu, Xi Shen, and Ying Shan. Generating human motion from textual descriptions with discrete representations. In *CVPR*, pp. 14730–14740, 2023a.

Jingyan Zhang, Han Liang, Ruichi Zhang, Bin Li, Juze Zhang, Xin Chen, Jingya Wang, Lan Xu, and Jingyi Yu. SCRIPT: Scalable diffusion policy with multi-stage training for language-driven physics-based humanoid control. *arXiv preprint arXiv:2605.22894*, 2026a.

Mingyuan Zhang, Xinying Guo, Liang Pan, Zhongang Cai, Fangzhou Hong, Huirong Li, Lei Yang, and Ziwei Liu. ReMoDiffuse: Retrieval-augmented motion diffusion model. In *ICCV*, 2023b.

Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. Motiondiffuse: Text-driven human motion generation with diffusion model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(6):4115–4128, 2024. doi: 10.1109/TPAMI.2024.3355414.

Xiaoyu Zhang, Steven Haener, Varun Madabushi, and Maegan Tucker. Kinodynamic motion retargeting for humanoid locomotion via multi-contact whole-body trajectory optimization. *arXiv preprint arXiv:2603.09956*, 2026b.

Yangsong Zhang, Anujith Muraleedharan, Rikhat Akizhanov, Abdul Ahad Butt, Gül Varol, Pascal Fua, Fabio Pizzati, and Ivan Laptev. PhysMoDPO: Physically-plausible humanoid motion with preference optimization. *arXiv preprint arXiv:2603.13228*, 2026c.

Zewei Zhang, Kehan Wen, Michael Xu, Junzhe He, Chenhao Li, Takahiro Miki, Clemens Schwarke, Chong Zhang, Xue Bin Peng, and Marco Hutter. Learning whole-body humanoid locomotion via motion generation and motion tracking. *arXiv preprint arXiv:2604.17335*, 2026d.

Zhikai Zhang, Jun Guo, Chao Chen, Jilong Wang, Chenghuai Lin, Yunrui Lian, Han Xue, Zhenrong Wang, Maoqi Liu, Jiangran Lyu, Huaping Liu, He Wang, and Li Yi. Track any motions under any disturbances. *arXiv preprint arXiv:2509.13833*, 2025b.

Qingrui Zhao, Kaiyue Yang, Xiyu Wang, Shiqi Zhao, Yi Lu, Xinfang Zhang, Qiu Shen, Xiao-Xiao Long, and Xun Cao. Make tracking easy: Neural motion retargeting for humanoid whole-body control. *arXiv preprint arXiv:2603.22201*, 2026.

Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *CVPR*, 2019.

Zifeng Zhuang, Ting Wang, Binghong Zou, Xinxin Luo, Jianfei Ma, Huaicheng Zhou, Jinxin Liu, and Donglin Wang. Humanoid-R0: Bridging text-to-motion generation and physical deployment via rl. OpenReview, withdrawn ICLR 2026 submission, 2025. URL https://openreview.net/forum?id=agohD5ewsR.

A Extended Related Work

Because of the main-paper page limit, its Related Work section gives a concise overview; this appendix provides a broader discussion of closely related work.

Text-to-motion generation. Text-to-motion generation maps natural language to human motion sequences and is commonly evaluated on HumanML3D and KIT-ML with retrieval, FID, diversity, and multimodality metrics (Plappert et al., 2016; Guo et al., 2022). Diffusion models such as MDM, MotionDiffuse, MLD, ReMoDiffuse, FlowMDM, and MotionLCM improve quality, controllability, retrieval augmentation, long-horizon composition, or sampling efficiency (Tevet et al., 2023; Zhang et al., 2024; Chen et al., 2023; Zhang et al., 2023b; Barquero et al., 2024; Dai et al., 2024). Discrete-token and language-model variants such as T2M-GPT, MotionGPT, MoMask, MMM, SATO, and HumanTOMATO improve sequence modeling, text alignment, and whole-body expressiveness (Zhang et al., 2023a; Jiang et al., 2023; Guo et al., 2024; Pinyoanuntapong et al., 2024; Chen et al., 2024; Lu et al., 2024). Their standard benchmarks measure semantic fidelity and motion diversity, but do not determine whether a generated reference is executable by a physical humanoid.

Language-conditioned humanoid motion. Recent work has begun to connect motion generation with humanoid execution. Kimodo scales controllable kinematic motion generation and releases robot-compatible assets (Rempe et al., 2026). TextOp, FRoM-W1, Humanoid-LLA, SafeFlow, MIND, and SCRIPT drive humanoids from language by combining high-level motion or intent generation with physics-based execution or action policies (Xie et al., 2026; Li et al., 2026b; Liu et al., 2025b; Cho et al., 2026; Li et al., 2026a; Zhang et al., 2026a). DreamControl-v2, PhyGile, CLAW, and RoboGhost further study robot-space priors, physics-prefix generation, language-annotated G1 data construction, or retargeting-free language-to-control interfaces (Harithas et al., 2026; Bao et al., 2026; Cao et al., 2026; Li et al., 2026e). GenTrack is closest to this emerging family, but targets a different experimental claim: a robot-native generator should improve under execution feedback, and the tracker should improve zero-shot coverage from the generated distribution rather than merely execute the generator's current outputs.

Joint generator–controller training. RobotMDM learns a motion-only critic that predicts the expected return of a frozen tracker, uses it as a differentiable loss to fine-tune a text-conditioned robot-motion diffusion model, and separately retrains trackers on generated motions (Serifi et al., 2024). PARC updates both components: it retrains the generator on tracker-corrected trajectories and the tracker on an expanded motion dataset (Xu et al., 2025). QuadFM alternates PPO controller updates with return-weighted generator updates and reconstruction supervision for text-conditioned quadruped control (Gao et al., 2026). Humanoid-DART retrains a goal-conditioned generator on relabeled elite trajectories and periodically retrains the humanoid tracker with PPO (Debbad et al., 2026). In contrast, PhysMoDPO constructs execution preferences using a fixed pretrained tracker and updates the generator with DPO (Zhang et al., 2026c). Other generator–tracker hierarchies improve a humanoid tracker online while freezing the generator (Zhang et al., 2026d). These are direct precedents for coupling the two components. Our scope differs in testing broad language-conditioned robot motion, online co-training, and held-out tracker coverage as separate claims under matched one-way and offline controls.

Retargeting and morphology. Human-to-humanoid retargeting maps SMPL, marker, video, or BVH motions to robot joint trajectories under morphology, contact, and joint-limit constraints. H2O and OmniH2O use retargeted human motion to train real-time humanoid teleoperation policies (He et al., 2024b;a). Recent retargeting studies such as General Motion Retargeting, implicit kinodynamic retargeting, kinodynamic trajectory optimization, physics-aware cross-morphology transfer, and neural motion retargeting show that retargeting quality strongly affects downstream tracking and controllability (Araujo et al., 2025; Chen et al., 2025; Zhang et al., 2026b; Huang et al., 2026; Zhao et al., 2026). In our experiments, retargeting is therefore not treated as an invisible preprocessing step: native G1 generation, held-out GMD-retargeted G1 references, and tracker execution failures are measured separately.

B Implementation Details

B.1 Robot Choice and Motion Representation

We use Unitree G1 as the primary robot morphology because it is an open humanoid robot model with ProtoMotions support, MJCF/URDF assets, pretrained tracker checkpoints, and a real-robot deployment path. This choice makes the robot-control claim concrete: success is measured on an actual humanoid morphology rather than on an SMPL-like physics proxy.

Robot-native motion representation. The generator emits a compact per-frame representation $\mathbf{q}_t \in \mathbb{R}^{38}$ that decodes exactly to a MuJoCo G1 configuration: a 3D root channel, a 6D continuous rotation (Zhou et al., 2019) of the pelvis orientation, and the 29 actuated joint angles. Two design choices make this target a deterministic function of language and easy for a flow-matching model to fit. *(i) Per-clip canonicalization.* The raw retargeted clips inherit an arbitrary world placement: the ground-plane (x, y) start position and the global heading (yaw about the up axis) are not determined by the caption, so a text-conditioned model regresses them toward the dataset mean and incurs metre-scale translation error. We therefore express every clip in a canonical frame whose first frame sits at the ground origin facing the canonical $+x$ direction, while preserving the physically meaningful height channel. *(ii) Displacement-based root.* The ground-plane translation is stored as a per-frame displacement rather than an absolute position; absolute position spans a large, sparse range (a run reaches several metres) that a flow-matching model systematically under-shoots, whereas the per-frame displacement is small and bounded, and the absolute trajectory is recovered by cumulative summation at decode time. The height channel stays absolute because it is bounded and semantically informative (e.g. a crawl versus a stand).

Offline retargeting as data construction. To obtain robot-native supervision, we retarget our internal human-motion dataset to the Unitree G1 skeleton *once, offline*, using the GMD retargeting pipeline. After internal quality control, this yields a paired training set $\mathcal{D}_{\text{pair}} = \{(\mathbf{c}_i, \mathbf{q}^{(i)})\}$ with 357,472 clips. We use these internal pairs only to initialize and rehearse G_θ; their GMD-retargeted trajectories are never supplied to tracker training. We warm-start G_θ from a pretrained human text-to-motion backbone, replace its output head with the 38-dimensional robot head, and supervise it on $\mathcal{D}_{\text{pair}}$. Crucially, retargeting is confined to this offline generator-training stage: at loop time the generator already speaks the robot's language, which removes the retargeting confound that otherwise contaminates any claim about generator physical realism.

Offline data curation. The generator training corpus is screened for scene compatibility and temporal quality. Curation removes motions requiring external objects or support geometry absent from the tracking environment, together with sequences containing temporal discontinuities or corrupted motion data. The resulting paired corpus supplies generator initialization, supervised rehearsal, and the training-prompt pool. These checks belong to offline data preparation; online reference admission operates on the exported generator outputs.

B.2 Generator Retention and Initialization

Execution reward alone can favor slow, low-amplitude motions that are easy to execute. Retention combines a KL penalty to the frozen initial generator during FlowGRPO with periodic supervised flow-matching rehearsal on the original train-only text–motion pairs. The rehearsal loss also includes a flow-velocity distillation term toward the same frozen generator. Generator prompts are sampled from the fixed curated training corpus. No TMR-G1, style, diversity, or prompt-coverage reward enters the training objective. Instead, TMR-G1 retrieval, MM-Dist, FID, and diversity are measured on a fixed held-out suite for model assessment.

Offline initialization. We build $\mathcal{D}_{\text{pair}}$ by retargeting our internal human-motion dataset to the G1 skeleton with GMD and fine-tune the HYMotion-based robot-native text-to-motion generator on these pairs to obtain G_0. We train this offline initialization once and reuse the same G_0 checkpoint across the five online post-training runs. Each co-training run updates its own generator and tracker; the five repetitions do not repeat offline generator initialization. This offline adaptation supplies a

motion source that follows diverse text conditions before online co-training begins. When a generator responds weakly to those conditions, sampling more prompts may fail to produce the requested range of behaviors, limiting the supervision available to the tracker. GenTrack can use an existing robot-native generator with suitable text following; training this initialization is our implementation choice. These pairs are used only for generator initialization and rehearsal and never enter tracker training. The tracker π_0 is initialized from an existing pretrained policy. We instantiate the framework separately with pretrained ProtoMotions and released SONIC trackers; their prior training is separate from GenTrack post-training. The initial tracker serves as the first lagged tracker; later checkpoints follow the update schedule below.

B.3 Tracker Curriculum and Generator Update Details

Reference admission and source mixture. Training prompts are drawn from the curated corpus. Generated candidates undergo integrity checks during export and scoring before entering the tracker reference pool. Processing failures and non-finite kinematic metrics are excluded. Admitted references include motions at different levels of current tracking performance, providing supervision for both maintaining learned skills and practicing challenging motions. Tracker updates sample from a merged pool of public and generated references. The evolving generator supplies the reference curriculum across rounds, while its execution reward promotes closer reference–execution agreement. Generated-motion and environment-transition budgets are matched to the offline controls. The generated-reference export pool has a configured retention cap of 13,337 motions. The packaged launch targets a generated fraction of 0.5 among tracker-snapshot entries by resampling generated references when needed. The tracker's native sampler and motion durations determine the resulting fraction of environment transitions from each source.

Tracker sampling. Reference admission determines the available training pool; the tracker's sampler determines which motions and segments are practiced. The SONIC reference configuration enables failure-based adaptive sampling over temporal bins. It mixes failure-weighted sampling probabilities with a 0.1 uniform component, allocating practice to difficult segments while retaining coverage of the motion library. This sampling mechanism operates within tracker training, separately from the generator's execution reward and reference admission.

Let c denote completion fraction, e_{joint} maximum joint error in radians, e_{traj} mean root-trajectory error in metres, and e_{disp} root-displacement error in metres. The implemented execution score is

$$\begin{aligned} S_{\text{exec}} = (1-c) + [e_{\text{joint}}]_2 + [e_{\text{traj}}/0.5]_2 \\ + 0.5[e_{\text{disp}}/0.5]_2 + 2\mathbb{I}_{\text{fall}}, \end{aligned} \tag{5}$$

where $[x]_2 = \min(x, 2)$ and $\mathbb{I}_{\text{fall}}$ is the scoring rollout's binary fall indicator. The lagged tracker remains fixed throughout each generator phase, while the current tracker has zero generator-reward weight. No additional velocity or amplitude gate is applied to this score.

For FlowGRPO, rewards are normalized within each same-prompt sample group and each freshly sampled trajectory group is replayed for four clipped policy-ratio updates. The full configuration uses a 0.02 KL weight to the frozen generator and a supervised rehearsal update of weight 1.0 every two FlowGRPO optimizer updates. Here the cadence counts FlowGRPO optimizer updates, including replay updates, rather than only newly sampled rollout groups.

Supervised rehearsal and reference implementation. For the normalized representation $\mathbf{x}_1$ of a training motion $\mathbf{q}$ and its text condition $\mathbf{c}$, sample $\mathbf{x}_0 \sim \mathcal{N}(0, I)$ and flow time $u \sim \mathcal{U}(0, 1)$, where I is the identity covariance matching the motion dimension. Set $\mathbf{x}_u = (1-u)\mathbf{x}_0 + u\mathbf{x}_1$ and let v_θ denote the generator's predicted flow velocity. Rehearsal minimizes

$$\mathcal{L}_{\text{rehearsal}} = \lambda_{\text{GT}}\mathbb{E}\big[\|v_\theta(\mathbf{x}_u, u, \mathbf{c}) - (\mathbf{x}_1 - \mathbf{x}_0)\|_2^2 + \lambda_{\text{dist}}\|v_\theta(\mathbf{x}_u, u, \mathbf{c}) - v_{\theta_0}(\mathbf{x}_u, u, \mathbf{c})\|_2^2\big], \tag{6}$$

with expectation over training pairs, noise, and flow time, and squared errors averaged over valid frames and channels. The overall rehearsal weight λ_{GT} and distillation weight λ_{dist} are both 1.0. The frozen flow predictor v_{θ_0} comes from G_0 and receives no gradient. This supervised-phase distillation is separate from the 0.02 KL penalty used during FlowGRPO.

The generator reference configuration specifies $K = 4$ generated candidates per prompt, 30 sampling steps, guidance 1.0, FlowGRPO noise parameter $\eta = 0.7$, and policy-ratio clipping 0.2. It uses

AdamW with learning rate 5×10^{-6}, betas $(0.9, 0.99)$, no weight decay or learning-rate scheduler, batch size 2 per GPU, gradient accumulation 1, FP32 arithmetic, and gradient-norm clipping at 1.0. These values specify the reference implementation. The packaged configuration inserts rehearsal every two FlowGRPO optimizer updates, following the protocol above.

Generator objective controls. The reward-weighted SFT control uses the same positive-advantage samples, while the Diffusion-DPO control forms preferences from the highest- and lowest-reward samples in each prompt group. Sampling and update budgets are matched across all three objectives.

Online co-training schedule. Each round first samples K generated candidates per curated training prompt. After reference admission, the lagged tracker evaluates their execution. Same-prompt rewards then create FlowGRPO advantages, and multi-epoch replay updates the generator with KL regularization and supervised rehearsal. Generated references accumulate in a merged pool with the fixed public references, and the tracker is updated by its method-native objective under a fixed total transition budget. The updated tracker becomes the next round's current tracker; the previous current tracker becomes its lagged tracker. Reward variance, policy-ratio movement, KL and supervised-rehearsal updates, exported references, and validation performance are checked at every round boundary.

Training budget and hardware. The online schedule comprises 12 rounds, with 120 generator training iterations and 150 tracker rollout/optimization iterations per round. Both generator and tracker training use eight NVIDIA H20 GPUs per phase. For the full generator objective, each round contains 80 FlowGRPO optimizer updates and 40 supervised rehearsal updates, totaling 960 and 480 updates, respectively, over training. The 1,440 total generator updates include replay and rehearsal. Distributed workers jointly perform each optimizer update; the update count is therefore independent of the number of workers.

Generated-reference budget. Under the full configuration, four FlowGRPO optimizer updates reuse each freshly sampled batch, giving $80/4 = 20$ new batches per round. Each batch contains two prompts per GPU on eight GPUs and $K = 4$ candidates per prompt. The schedule therefore specifies 320 prompt-sampling occurrences and $20 \times 2 \times 8 \times 4 = 1,280$ new generated candidates per round, or 15,360 candidates across the 12 rounds of one online run. Repeated prompts count as separate sampling occurrences. These are pre-admission generation budgets; admitted references are the subset successfully processed and exported by the integrity checks. Replay updates and supervised rehearsal do not add new candidates to this count.

For ProtoMotions, each tracker iteration collects 32 environment steps from each of 1,024 environments per GPU. Across eight GPUs, the 150 iterations in a round collect $150 \times 32 \times 1{,}024 \times 8 = 39{,}321{,}600$ transitions. The complete 1,800-iteration schedule thus specifies 471,859,200 tracker transitions. This count covers tracker training; generator-scoring rollouts contribute to the generator phase's simulation cost. PPO minibatch reuse performs additional optimization on the collected transitions. Matched tracker controls receive the same environment-transition budget within each backbone. Generator updates, generated candidates, and tracker transitions are recorded as distinct quantities.

SONIC uses 4,096 environments per GPU and a 24-step rollout horizon. With eight GPUs and 150 iterations per round, it collects $150 \times 24 \times 4{,}096 \times 8 = 117{,}964{,}800$ transitions per round, or 1,415,577,600 over 12 rounds. The different rollout sizes give each backbone its own transition budget; comparisons within a backbone match that budget.

C Detailed Experimental Protocol

Robot and motion representation. All experiments use the Unitree G1 morphology and the canonical 38-dimensional motion representation described above. References, generated motions, executed trajectories, and visualizations are exported at 30 FPS. Matched training comparisons use the same tracker architecture, simulator, optimizer, initialization, and environment-transition budget. External trackers are run with their released implementations, but their outputs are converted to the same start-aligned G1 reference/execution format before root-aware tracker evaluation. Their native

Table 3: Robot-native source validation on the fixed held-out test suite. Succ. and Root err. are measured by executing each reference with the same frozen IsaacLab SONIC policy. Succ. is recomputed from complete 30-FPS exports with the shared fall-only protocol; released callback flags are diagnostic only. Ref. skate is computed on each reference with the same MuJoCo-FK contact protocol. Semantic metrics use TMR-G1, with MM-Dist and Diversity in raw latent space and FID in unit-normalized latent space. Succ. is a percentage, Root err. is in metres, and Ref. skate is in metres per second. Bold marks the best value per metric; Diversity is descriptive.

Source	Succ. (%) ↑	Root err. ↓	Ref. skate ↓	TMR-G1 R@1 ↑	MM-Dist ↓	FID ↓	Diversity
Retargeted GT	91.41	0.402	0.126	**0.849**	**16.345**	**0.000**	36.713
KIMODO-G1 (Rempe et al., 2026)	**98.63**	**0.101**	**0.087**	0.500	25.215	0.122	35.689
HYMotion-based (G_0)	92.58	0.381	0.119	0.774	18.293	0.023	36.375

success flags are not used for tracker comparison. Generator executability is evaluated separately with the frozen SONIC executor and the low-level metric family defined below. The evaluation checkpoint is kept fixed across all generator variants and is not adapted to their generated motions. This frozen executor is shared by every generator row and provides a consistent executability measure under frozen SONIC execution on simulated G1.

Training data. The generator-only internal corpus contains 357,472 GMD-retargeted G1 training pairs and a fixed held-out test set. Training and evaluation motions are separated at data collection, with no duplicate or near-duplicate motions shared across the partitions. This separation concerns the underlying motion content and is maintained through preprocessing and retargeting. We apply an internal curation and quality-control protocol within these fixed partitions. Across tracker post-training arms, the only admissible sources are the public pool of 12,733 AMASS-G1 and 604 LAFAN1-G1 motions and G1 references sampled from the relevant generator; reference-only continuation uses the public pool alone. Captions from the internal training split may provide generator prompts, but their paired GMD-retargeted trajectories are never provided to the tracker.

Tracker protocols and baselines. We instantiate GENTRACK from two existing tracker checkpoints: a root-aware ProtoMotions/AMP-PPO checkpoint (NVLabs, 2025) and the official released SONIC checkpoint (Luo et al., 2025). For each backbone, the primary post-training arms start from the same checkpoint hash, use the same simulator and optimizer, match the environment-transition budget, and share validation and test manifests. The primary comparison is the frozen checkpoint versus equal-budget reference-only continuation, frozen-G_0 replay, and GENTRACK. In Table 1, Ref. only denotes reference-only continuation and G_0 replay abbreviates frozen-G_0 replay. We additionally report G_{final} replay, using the final online generator. This control restarts from the same tracker checkpoint and tests replay from the final generated distribution. Differences from the online run can also reflect curriculum and optimization history. These are post-training comparisons, not evidence for training from scratch. Released Any2Track (Zhang et al., 2025b), Humanoid-GPT (Qi et al., 2026), SONIC, and BeyondMimic (Liao et al., 2025) are external references; their weights are not used to initialize either adapting branch, except for the explicitly labeled SONIC branch itself.

Evaluation splits. We evaluate zero-shot tracking on three fixed, mutually disjoint G1 splits. LAFAN1-G1 contains 40 locomotion and transition sequences (Harvey et al., 2020); AMASS-test-G1 contains 138 motions following the protocol used by prior tracking work (Mahmood et al., 2019; NVLabs, 2025); and Wild-G1-clean contains 1,024 motions from an independent, non-public data source and is curated for compatibility with the flat-ground evaluation setup. Wild-G1-clean has no overlap with LAFAN1, AMASS, or any other public open-source motion dataset. Its test motions are also disjoint from the internal motion data used for generator initialization and supervised rehearsal. The exact source is not disclosed. Test motions and their paired text are excluded from generator initialization, supervised rehearsal, GenTrack post-training, replay construction, and TMR-G1 training. This held-out protocol does not by itself establish non-overlap with the complete pretraining history of each released checkpoint.

Table 4: Metric definitions and evaluation roles. Generator metrics evaluate text-conditioned references with frozen-SONIC execution and TMR-G1, while tracker metrics evaluate policies executing fixed references. Low-level generator metrics follow the RLPF structure, and tracker success follows SONIC's relaxed criterion recomputed from shared 30-FPS G1 trajectories. Auxiliary diagnostics are listed for their intended roles; entries without reported results do not provide evidence of difficulty preservation or scoring-tracker calibration.

Metric	Definition	Used for	Direction
Generator Succ.	fraction of generated references whose complete 30-FPS official SONIC execution maintains reference-relative pelvis-height deviation at most 0.25 m (Luo et al., 2025; Yue et al., 2025)	generator physical quality	↑
E_{joint}	mean wrapped rotation error over the 29 actuated G1 joints between reference and execution, in radians	generator physical quality	↓
E_{key}	mean start-XY-aligned position error over SONIC's fixed official 14 tracked bodies, in metres	generator physical quality	↓
Fall-only SR	fraction of unseen references whose complete 30-FPS rollout maintains maximum reference-relative pelvis-height deviation at most 0.25 m; one missing resampled endpoint is tolerated (Luo et al., 2025)	tracker zero-shot coverage	↑
Completion / Unexpected Fall	executed-frame ratio and persistent reference-relative collapse when reference pelvis height is at least 0.50 m and up-axis cosine is at least 0.50; low-floor targets are excluded and absolute execution height is never used	auxiliary diagnostics	mixed
MPJPE	mean Cartesian position error over SONIC's official 14-body set after subtracting each frame's pelvis translation, in millimetres; average valid frames within each motion, then weight motions equally across all splits (Luo et al., 2025)	tracker pose fidelity	↓
E_g	start-XY-aligned global Cartesian position error over the same 14-body set, in millimetres; the same all-trajectory aggregation as MPJPE	tracker global fidelity	↓
MPJVE	mean finite-difference Cartesian velocity error over the same root-relative 14-body trajectories, in metres per second; the same all-trajectory aggregation as MPJPE	tracker motion fidelity	↓
RootVelErr	mean Cartesian linear-velocity error of the robot root, in metres per second; the same all-trajectory aggregation as MPJPE	tracker trajectory fidelity	↓
Root err.	mean start-aligned root-trajectory position error between reference and execution	physical feasibility	↓
Foot skating / penetration	contact-frame foot speed and below-ground penetration depth computed on generated references or physics rollouts	physical artifacts	↓
Robot kinematic diagnostics	joint-limit saturation, contact-rhythm mismatch, root/yaw statistics, and other robot-centric pose or velocity checks; these are auxiliary diagnostics rather than main distribution metrics	morphology compatibility audit	mixed
Prompt cov. / action entropy	prompt/action coverage and action-category entropy over the fixed generator prompt suite	anti-degeneration	↑
Scoring-tracker calibration	margin by which lagged checkpoints from the adapting tracker branch rank held-out GT references above corrupted or invalid references before their reward is used for generator optimization	reward reliability	↑
TMR-G1 metrics	TMR-G1 R-Precision and raw-latent MM-Dist/diversity, together with unit-normalized-latent FID, on the fixed robot-motion suite	semantic/distribution preservation	mixed

C.1 Metric Definitions and Computation

Tracker metrics. Every tracker is first rolled out until the reference motion timeout with its method-native failure termination disabled. We export reference and execution states, resample each trajectory once to 30 FPS, and apply the same post-hoc termination test to every method. Following SONIC's cross-method MuJoCo comparison (Luo et al., 2025), a trajectory fails only if its maximum reference-relative pelvis-height error exceeds 0.25 m. A single missing endpoint sample caused by resampling is tolerated. Because the height test is reference-conditioned, a commanded fall or get-up is not rejected merely for approaching the ground. The released-code evaluator's additional ankle/wrist height and pelvis-orientation terms, method-native success flags, and Any2Track's mean-error criterion are retained only as diagnostics and never enter the main table.

All simulator evaluations start from reference frame zero. We disable observation corruption, reset pose/velocity perturbations, adaptive motion sampling, and startup randomization of friction, restitution, default joint positions, center of mass, and body mass. These choices make the nominal initial state deterministic across checkpoints.

Using SONIC's root-relative position definition and the position–velocity metric family used by Humanoid-GPT (Luo et al., 2025; Qi et al., 2026), we report mean per-joint position error (MPJPE), start-XY-aligned global joint position error (E_g), mean per-joint velocity error (MPJVE), and root linear-velocity error (RootVelErr). Because the source papers use different representations and units, all four are recomputed in Cartesian robot space from the same synchronized trajectories. MPJPE subtracts the pelvis translation from each pose, evaluates SONIC's fixed 14-body set, and is reported in millimetres. E_g retains global joint positions after start-XY alignment and is reported separately in millimetres. MPJVE applies the same root-relative alignment and body set to finite-difference velocities (m/s), while RootVelErr measures global pelvis linear-velocity error (m/s).

For each of the four errors, we first average valid frames (or finite-difference intervals) within each motion, then average equally over evaluated motions, including failed trajectories. Overall errors combine all 1,202 motions; the three splits therefore have weights $40/1202$, $138/1202$, and $1024/1202$. We compute each metric within a run and then average over five runs. For trained components, superscripts report two-sided 95% Student-t confidence-interval half-widths, $h = t_{0.975,4}s/\sqrt{5}$, where s is the sample standard deviation of the five run-level metrics and $t_{0.975,4}$ is the 97.5th percentile of Student's t distribution with four degrees of freedom. Overall-error intervals are computed after combining the splits within each run. These intervals condition on the shared offline initialization and fixed evaluation set. Fixed-checkpoint and unchanged-component entries have no across-training-run interval. The intervals quantify uncertainty in each method's mean; relative comparisons summarize observed mean differences, rather than intervals for those differences. Training repetitions refer to online post-training from the shared initialization. Fall-only SR is shown alongside the errors as the separate coverage statistic. Completion, reference-conditioned Unexpected Fall, and acceleration error remain auxiliary diagnostics and do not enter the success decision or the main table.

Per-split tracking accuracy. Table 5 separates local pose and global position errors for the SONIC branch. Relative to initialization, GenTrack lowers mean MPJPE on all three splits and mean E_g on LAFAN1-G1 and Wild-G1-clean; AMASS-test-G1 E_g increases from 389.2 to 396.2 mm. Relative to both replay controls, GenTrack has lower mean errors on every split. The observed mean gains also extend to the two public splits, with a global-position trade-off against initialization on AMASS-test-G1.

Generator metrics. Following the high-level/low-level evaluation decomposition of RLPF (Yue et al., 2025), generator quality is measured by semantic generation metrics and by official frozen-SONIC execution in IsaacLab. We report success rate Succ, mean wrapped actuated-joint rotation error E_{joint} (rad), and mean start-XY-aligned position error E_{key} (m) over SONIC's fixed official 14 tracked bodies. We use clearer notation because RLPF's "MPJPE (rad)" is a joint-rotation rather than a Cartesian position error. Succ is recomputed from complete 30-FPS exports using the same fall-only, reference-conditioned termination term as the tracker table. The frozen policy, motion-timeout horizon, reference, and start alignment are held fixed. Reported durations and continuous errors are computed from complete 30-FPS exports rather than simulator-specific early-reset flags.

Table 5: Per-split SONIC tracking errors (mm, lower is better), averaged over five online runs for trained configurations. Each motion contributes equally within its split, including failed trajectories. Split sizes are 40, 138, and 1,024 motions, respectively. Bold marks the lowest mean error in each column. Superscripts give ±95% CI half-widths across five runs; initialization is a fixed checkpoint.

	LAFAN1-G1		AMASS-test-G1		Wild-G1-clean	
SONIC configuration	MPJPE	E_g	MPJPE	E_g	MPJPE	E_g
Initialization	40.1	482.7	46.5	**389.2**	140.3	884.4
G_0 replay	$43.2^{\pm 2.7}$	$503.8^{\pm 11.7}$	$49.3^{\pm 2.4}$	$421.6^{\pm 14.0}$	$148.1^{\pm 3.1}$	$918.4^{\pm 15.4}$
G_{final} replay	$39.7^{\pm 2.3}$	$478.9^{\pm 11.0}$	$47.6^{\pm 2.4}$	$408.3^{\pm 14.7}$	$140.2^{\pm 2.7}$	$913.5^{\pm 13.1}$
GENTRACK	$\mathbf{38.6}^{\pm 2.4}$	$\mathbf{466.4}^{\pm 16.6}$	$\mathbf{45.9}^{\pm 2.3}$	$396.2^{\pm 13.5}$	$\mathbf{138.0}^{\pm 3.1}$	$\mathbf{875.9}^{\pm 20.5}$

Unexpected Fall, root error, foot skating, and penetration remain auxiliary diagnostics rather than extra success gates.

TMR-G1 supplies R-Precision (R@1/2/3), MM-Dist, FID, and diversity. MM-Dist and diversity use raw TMR-G1 latents, whereas FID uses unit-normalized latents to avoid scale-dominated Fréchet values. SMPL- or HumanML3D-space evaluators are not used for the main generator result.

TMR-G1 implementation and retrieval protocol. TMR-G1 is a G1-native reproduction of TMR (Petrovich et al., 2023), with text and motion encoders and a motion reconstruction decoder. Its reference artifact specifies a 256-dimensional latent, six Transformer layers, four attention heads, a feed-forward width of 1,024, dropout 0.1, and GELU activation. Text inputs use 768-dimensional DistilBERT token features. Motion inputs are canonicalized 30-FPS, 38-dimensional G1 sequences, standardized using the artifact's stored mean and standard deviation. Variable-length sequences are padded with valid-frame masks. Both encoders run in evaluation mode and use the latent posterior mean rather than stochastic posterior samples.

The evaluator release metadata records 359,153 clips, split into 344,966 training, 7,083 validation, and 7,104 test clips, and an epoch-139 checkpoint. These are evaluator-training metadata, distinct from the generator-training corpus size reported above and the 1,024-prompt generation evaluation suite. The generation evaluation pairs are drawn from the held-out test splits described above. Neither their reference motions nor their paired text are used to train TMR-G1. The same frozen TMR-G1 encoders evaluate every generator variant on these common test pairs.

The reference evaluation script uses a seeded permutation (seed 0), retrieval groups of 32, and one evaluation repeat. For R@1, R@2, and R@3, motion embeddings are ranked by Euclidean distance from the paired text embedding within each group; the paired motion is the single positive. Thus the retrieval candidate pool is 32, not the full 1,024-sample suite. For that suite, all 32 complete groups are evaluated. MM-Dist averages paired text–motion distances in the original latent space. Diversity averages distances between 300 randomly sampled embedding pairs, using two independently sampled index sets without replacement within each set. FID compares the means and covariances of the full reference and generated embedding sets after per-sample L2 normalization. An evaluation repeat changes retrieval grouping and diversity sampling; it is not an independent training run. All generator variants must share the evaluator, sample ordering, and randomization protocol.

Evaluation assets. The anonymous supplementary code includes the frozen epoch-139 TMR-G1 weights, matching 38-dimensional normalization statistics, architecture and preprocessing configurations, and a command-line evaluator. These assets load from the packaged directory. The text frontend uses the public `distilbert-base-uncased` model. The supplement also includes the physical-metric implementation and input schemas for saved reference and execution trajectories. Private motion records and the evaluation trajectories are not distributed in the archive.

C.2 ROBOT-NATIVE SOURCE VALIDATION

Table 3 evaluates the held-out G1 references and KIMODO-G1 (Rempe et al., 2026) under the same prompts and evaluators. The GT physical columns are obtained by executing the retargeted refer-

Table 6: Matched closed-loop ablations from the ProtoMotions initialization. Panel (a) separates generator and tracker update directions; panel (b) compares the full configuration with variants that change only the execution feedback, scoring-tracker timing, or generator objective. Tracker only freezes G_0, Generator only freezes π_0, and GENTRACK uses FlowGRPO with a lagged tracker. Current tracker uses the current tracker for scoring, and RW-SFT denotes reward-weighted SFT. All variants share prompts, generated-candidate groups, accepted-motion and tracker-transition budgets, validation, and frozen evaluators. Generator metrics use the fixed held-out prompt suite. Bold marks the best mean within each panel, including ties. Split headers follow Table 1. Superscripts give ±95% CI half-widths across five runs.

Variant	Tracker fall-only SR (%) ↑			Generator				
	LAFAN1	AMASS-test	Wild-G1	Succ. (%) ↑	TMR-G1 R@1 ↑	E_{joint} (rad) ↓	E_{key} (m) ↓	TMR-G1 FID ↓
(a) Update directions								
No update (G_0, π_0)	75.0	**81.2**	45.9	92.58	0.774	0.159	0.410	0.023
Tracker only	**77.5**$^{\pm 2.2}$	79.0$^{\pm 1.4}$	46.8$^{\pm 0.9}$	92.58	0.774	0.159	0.410	0.023
Generator only	75.0	**81.2**	45.9	**93.95**$^{\pm 0.68}$	**0.783**$^{\pm 0.005}$	**0.157**$^{\pm 0.002}$	**0.381**$^{\pm 0.005}$	**0.020**$^{\pm 0.002}$
GENTRACK	75.0$^{\pm 3.1}$	**81.2**$^{\pm 1.8}$	**47.3**$^{\pm 1.2}$	93.55$^{\pm 0.81}$	0.782$^{\pm 0.005}$	0.160$^{\pm 0.004}$	0.399$^{\pm 0.009}$	**0.020**$^{\pm 0.003}$
(b) Feedback and objective								
w/o exec. reward	**77.5**$^{\pm 2.2}$	79.0$^{\pm 1.3}$	46.1$^{\pm 0.9}$	92.71$^{\pm 0.56}$	0.778$^{\pm 0.004}$	0.161$^{\pm 0.001}$	0.408$^{\pm 0.009}$	0.024$^{\pm 0.002}$
Success only	**77.5**$^{\pm 3.1}$	80.3$^{\pm 1.7}$	46.7$^{\pm 1.4}$	93.12$^{\pm 0.77}$	0.780$^{\pm 0.003}$	0.160$^{\pm 0.002}$	0.406$^{\pm 0.011}$	0.021$^{\pm 0.004}$
Current tracker	75.0$^{\pm 4.9}$	79.6$^{\pm 2.5}$	46.4$^{\pm 1.4}$	93.01$^{\pm 0.95}$	0.777$^{\pm 0.003}$	0.163$^{\pm 0.005}$	0.412$^{\pm 0.011}$	0.026$^{\pm 0.005}$
RW-SFT	**77.5**$^{\pm 3.1}$	79.9$^{\pm 1.5}$	46.6$^{\pm 1.0}$	**94.10**$^{\pm 0.67}$	0.769$^{\pm 0.005}$	**0.158**$^{\pm 0.004}$	**0.389**$^{\pm 0.008}$	0.029$^{\pm 0.002}$
Diffusion-DPO	75.0$^{\pm 3.1}$	80.6$^{\pm 2.0}$	46.9$^{\pm 1.1}$	93.36$^{\pm 0.70}$	0.776$^{\pm 0.003}$	0.159$^{\pm 0.005}$	0.402$^{\pm 0.008}$	0.023$^{\pm 0.001}$
GENTRACK	75.0$^{\pm 3.1}$	**81.2**$^{\pm 1.8}$	**47.3**$^{\pm 1.2}$	93.55$^{\pm 0.81}$	**0.782**$^{\pm 0.005}$	0.160$^{\pm 0.004}$	0.399$^{\pm 0.009}$	**0.020**$^{\pm 0.003}$

Table 7: Matched generator-retention ablations for the ProtoMotions branch. λ_{KL} weights the frozen-generator penalty and λ_{GT} weights the supervised rehearsal update applied every two Flow-GRPO optimizer updates in the full configuration. The controls remove either retention component or both while retaining the prompts, online-loop schedule, optimization budget, and frozen evaluators. Bold marks the strongest mean physical value or the best-preserved mean semantic/distribution value. Superscripts give ±95% CI half-widths across five runs.

Variant	λ_{KL}	λ_{GT}	Succ. (%) ↑	E_{key} (m) ↓	TMR-G1 R@1 ↑	TMR-G1 FID ↓	Diversity
w/o KL	0	1.0	93.85$^{\pm 0.71}$	0.389$^{\pm 0.010}$	0.773$^{\pm 0.005}$	0.028$^{\pm 0.004}$	36.102$^{\pm 0.245}$
w/o rehearsal	0.02	0	94.08$^{\pm 0.83}$	0.382$^{\pm 0.014}$	0.768$^{\pm 0.005}$	0.034$^{\pm 0.004}$	35.812$^{\pm 0.227}$
w/o both	0	0	**94.63**$^{\pm 1.06}$	**0.369**$^{\pm 0.016}$	0.749$^{\pm 0.006}$	0.051$^{\pm 0.004}$	34.684$^{\pm 0.287}$
GENTRACK	0.02	1.0	93.55$^{\pm 0.81}$	0.399$^{\pm 0.009}$	**0.782**$^{\pm 0.005}$	**0.020**$^{\pm 0.003}$	**36.373**$^{\pm 0.261}$

ences with the frozen tracker, rather than by a trivial reference-to-reference comparison. KIMODO-G1 achieves higher execution success (98.63% versus 92.58%), while our initialized G_0 has stronger text–motion alignment on this suite: TMR-G1 R@1 is 0.774 versus 0.500, and MM-Dist is 18.293 versus 25.215. This comparison motivates using G_0 as the text-conditioned supervision source and subsequently refining its execution quality through the online loop.

D ADDITIONAL EXPERIMENTS AND ABLATIONS

D.1 CONTROL EXPERIMENTS

The controls in the main tables cover reference-only continuation, tracker-only frozen-G_0 replay, tracker-filtered SFT, and frozen-SONIC FlowGRPO. Together, they provide direct comparisons for additional tracker optimization, offline replay, and generator feedback. Table 6 completes the matched ProtoMotions comparison by separating update directions in panel (a) and varying the execution score, scoring tracker, and generator objective in panel (b).

Generator-side effect. Relative to G_0 in the main-paper generator table, generator-only feedback raises mean frozen-SONIC execution success from 92.58 to 93.95 and reduces E_{key} from 0.410 to 0.381. At the same time, E_{joint} decreases from 0.159 to 0.157, while TMR-G1 R@1 and FID change from 0.774/0.023 to 0.783/0.020. The frozen-SONIC FlowGRPO control improves E_{key} to 0.363 but lowers success and R@1 to 90.92/0.767 and raises FID to 0.027, whereas the online SONIC branch reaches 94.43 success, $E_{\text{key}} = 0.325$, R@1 $= 0.783$, and FID $= 0.020$. Within ProtoMotions, however, generator-only training has more favorable means for success, joint/key-body error, and R@1 than the full loop, with identical mean FID. The key-body error intervals are separated, whereas success, joint-error, and R@1 intervals overlap. The results therefore support branch-dependent outcomes rather than uniform improvement from updating both components.

Tracker-side effect. The tracker controls assess the value of updating generated supervision online. Frozen-G_0 replay produces split-dependent mean changes. With SONIC, the full loop raises its three mean success rates from 85.0/78.3/45.5 to 90.0/79.7/48.0 relative to frozen-G_0 replay, while reducing MPJPE from 133.3 to 124.1 mm and E_g from 847.6 to 807.2 mm; the position-error intervals are separated in both comparisons. With ProtoMotions, its means favor AMASS-test and Wild-G1-clean success and MPJPE. Reference-only continuation has a different cross-split mean pattern. ProtoMotions also has lower mean LAFAN1 SR and higher mean global and velocity errors than frozen-G_0 replay. These comparisons characterize coverage–fidelity trade-offs; they do not uniquely separate mutual adaptation from curriculum and optimization history.

Execution-feedback and objective controls. Panel (b) of Table 6 maps each comparison directly to a method choice through five-run means and their uncertainty. Removing execution reward leaves the generator at 92.71 success, 0.778 R@1, and 0.024 FID, while AMASS-test/Wild-G1-clean SR reach 79.0/46.1. Success-only feedback improves these values to 93.12/0.780/0.021 and 80.3/46.7, respectively. The dense-score configuration has more favorable means, with overlapping intervals for these comparisons. Using the current tracker for scoring yields mean cross-split SR of 75.0/79.6/46.4 and $E_{\text{joint}}/E_{\text{key}}$ of 0.163/0.412. Reward-weighted SFT obtains the highest mean success rate (94.10) and lowest mean $E_{\text{joint}}/E_{\text{key}}$ (0.158/0.389), but reduces R@1 to 0.769 and raises FID to 0.029. Diffusion-DPO provides a more balanced control (93.36 success, 0.776 R@1, 0.023 FID) and reaches 80.6/46.9 on AMASS-test/Wild-G1-clean. FlowGRPO has more favorable means for execution success, key-body error, R@1/FID, and coverage on both target splits, with overlapping intervals. These matched comparisons characterize how feedback and objective choices balance coverage, execution fidelity, and motion-content metrics.

Generator-retention controls. Table 7 connects the preservation mechanisms to the generator-side claim. Removing either retention component raises mean execution success but reduces TMR-G1 retrieval, increases FID, and lowers diversity; the degradation is largest when both retention components are removed. The full configuration therefore trades mean execution performance for better retrieval, FID, and diversity. The clearest separation occurs after removing both components: the R@1, FID, and diversity intervals are separated from the full configuration, whereas execution-success intervals overlap. The table does not separately isolate the flow-velocity distillation term within rehearsal. This ablation does not diagnose the cause of the tracker-filtered SFT results.

D.2 Qualitative and Distribution-Gap Analysis

Paired qualitative evidence. The paired visualizations expose the complementary effects targeted by our alternating updates. Under the same prompt, length, and sampling noise, replacing G_0 with the GenTrack generator turns a reference that the same frozen SONIC policy fails to execute into one that it completes. Conversely, for the shown held-out Wild-G1-clean reference, both the initial tracker and the frozen-G_0 replay control fail, whereas the online GenTrack tracker completes the motion. Together with the main-paper generator and tracker tables, these cases visualize the role of each branch: tracker rollouts guide the generator toward robot-compatible references, while an evolving generated-reference curriculum extends tracker coverage beyond offline replay. All snapshots render logged poses on the Unitree G1 STL model.

Motion-to-execution gap diagnostics. The main-paper motion-to-execution gap figure analyzes a diagnostic high-fidelity subset selected by predefined execution-fidelity thresholds. Shared PCA

of 9,332 paired windows exhibits broad reference–execution overlap, with its first two components explaining 61.1% of pooled variance. The shared structure supports robot-native initialization and rehearsal, but a systematic local gap remains: a motion-case-grouped classifier in the full descriptor space reaches an AUC of 0.996 ± 0.004, compared with 0.505 ± 0.006 under shuffled labels. Across all 10,369 diagnostic windows, descriptor densities localize the largest shifts to body jerk, foot skating, root height, and foot clearance, with a smaller change in peak root speed. These results support the central method choice to evaluate motions through closed-loop execution and use a dense trajectory-level score, rather than treating robot-space references or a fixed generated pool as executable by construction.