DREAMPOLICY: SCALING HUMANOID LOCOMOTION THROUGH TERRAIN-AWARE GENERATIVE PLANNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Achieving versatile humanoid locomotion with a single policy remains a key scalability challenge. Existing approaches often distill multiple terrain-specific policies into a unified policy, but such distillation mainly transfers individual locomotion primitives and provides limited ability to compose them when encountering novel and structurally different terrains. We present DreamPolicy, a scalable framework for humanoid locomotion through terrain-aware generative planning. We first construct a modular multi-terrain benchmark with over a dozen primitive terrains and their combinations, and collect 75M transitions of expert rollouts from specialized policies, providing diverse locomotion primitives and terrain-conditioned behaviors for learning. On this dataset, we train a terrain-aware autoregressive diffusion planner to generate a humanoid motion imagination (HMI), a physically plausible future state trajectory conditioned on state history and terrain information. A unified HMI-conditioned policy then tracks these generated future states, allowing locomotion behaviors to be dynamically composed according to the encountered terrain without terrain-specific reward design. DreamPolicy scales consistently with offline data and generalizes beyond the terrains covered by the source policies. Experiments show substantial improvements over strong baselines on unseen and composite terrains, together with zero-shot transfer to a Unitree G1. These results demonstrate that terrain-aware generative planning provides a scalable approach to humanoid locomotion beyond the "one task, one policy" regime.

1 INTRODUCTION

The rapid progress of embodied intelligence has enabled legged robots to perform increasingly dynamic behaviors that require balance, perception, and adaptive control (Vukobratović & Stepanenko, 1972; Xie et al., 2025). Quadrupedal robots have demonstrated impressive capabilities in challenging locomotion tasks such as parkour (Zhu et al., 2026a; Cheng et al., 2024c; Zhuang et al., 2023; Zargarbashi et al., 2024; Fu et al., 2021) and whole-body loco-manipulation (Fu et al., 2022; Liu



Figure 1: We propose DreamPolicy, a scalable unified policy that leverages a diffusion-based planner to enable versatile humanoid locomotion across diverse terrains.

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.

et al., 2024b). Humanoid robots further offer an anthropomorphic morphology naturally suited to human-centric environments, with recent advances spanning locomotion (Wang et al., 2026; Zhuang et al., 2024; Radosavovic et al., 2024a;b), manipulation (Chen et al., 2025; Gu et al., 2025), and whole-body loco-manipulation (Cheng et al., 2024b;a; Fu et al., 2024; He et al., 2024c; Xue et al., 2025). Despite this progress, most existing systems remain specialized: policies that work well on individual terrain types often struggle on diverse or previously unseen terrain combinations. Developing a *single* humanoid policy that robustly traverses such heterogeneous environments therefore remains a fundamental scalability challenge.

A growing body of work improves humanoid control by incorporating pre-trained models (Zhang et al., 2025b; Albaba et al., 2025) or human motion priors (Zhuang et al., 2026; Wu et al., 2026; Zhang et al., 2024; Fu et al., 2024; He et al., 2024b; Jiang et al., 2024), which constrain policy optimization toward feasible motions and reduce the exploration burden of reinforcement learning. However, adapting these priors to diverse terrain structures—such as slopes, gaps, stairs, and uneven surfaces—and to external disturbances (Jenelten et al., 2019; Radosavovic et al., 2024b; Wang et al., 2025; Zhuang et al., 2024; Ren et al., 2025) remains challenging. Moreover, learning-based policies rely on terrain-specific observations, curricula, and hand-designed rewards, making them prone to overfitting to the training environments (Gu et al., 2024; Long et al., 2024). Policy distillation (Zhuang et al., 2023) instead consolidates specialized policies into a single student, but provides no explicit mechanism for recomposing their primitives over time on novel terrains, so its generalization remains bounded by the teacher policies' terrain distributions.

This observation motivates a different perspective: instead of compressing terrain-specific controllers into a monolithic policy, can we learn a *generative model of terrain-conditioned future motion* to guide a unified controller? Such a model would accumulate locomotion knowledge from scalable offline experience rather than ever more complex reward engineering, and would synthesize future motion online as an explicit intermediate representation for recomposing locomotion primitives. The central challenge is collecting sufficiently diverse terrain-conditioned experience and learning a planner that generalizes beyond the terrains it was collected from.

To this end, we present DreamPolicy, a scalable framework for humanoid locomotion through terrain-aware generative planning. We first build a modular multi-terrain benchmark and aggregate 75M expert transitions from specialized policies. These transitions are not merely demonstrations for distillation: they expose a terrain-aware autoregressive diffusion planner to diverse locomotion primitives and terrain-conditioned state transitions. The planner generates an HMI, a physically plausible future state trajectory conditioned on state history and terrain information, which serves as an explicit, dynamically updated locomotion objective rather than an abstract latent representation. A unified HMI-conditioned policy then tracks these imagined states with motion-manifold regularization, amortizing terrain-specific reward shaping into the expert-data generation stage while adapting online to the encountered terrain. As offline experience grows, the planner observes a richer distribution of terrain-conditioned transitions without any increase in the number of deployed policies.

Our main contributions are summarized as follows:

- **Modular Terrain Benchmark and Large-Scale Primitive Skill Collection.** We construct a modular humanoid locomotion benchmark with more than a dozen primitive terrains and their combinations, and collect 75M expert transitions from specialized policies trained on five representative terrains. This dataset provides diverse locomotion primitives and terrain-conditioned transitions for learning scalable generative locomotion models.

- **Terrain-Aware Generative Planning for Unified Humanoid Control.** We introduce a terrain-aware autoregressive diffusion planner that generates an HMI from state history and terrain information, and a unified HMI-conditioned policy that tracks these generated states with motion-manifold regularization. This shifts terrain-specific reward shaping into the expert-data generation stage and enables online composition of locomotion behaviors.

- **Scaling and Generalization to Unseen Terrains.** We show that DreamPolicy improves consistently as offline experience scales and generalizes beyond the terrains represented by its source policies: it outperforms the strongest baseline by up to 27% on unseen terrains and 38% on combined terrains, and transfers zero-shot to a Unitree G1, improving real-world success rates from 40–45% to 80–85% on terrains requiring precise foot placement.

2 Related Work

2.1 Humanoid Locomotion and Generalist Control

Recent advances in humanoid robotics have predominantly relied on hierarchical reinforcement learning, where methods like Extreme Parkour (Cheng et al., 2024c) and Distillation-PPO (Zhang et al., 2025a) employ specialized skills for robust locomotion. To bridge the Sim2Real gap, works such as DWL (Gu et al., 2024) and Humanoid Perception Controller (Sun et al., 2025) focus on latent state reconstruction. However, these approaches often suffer from heavy reliance on manual reward engineering. To address this, data-driven methods like H2O (He et al., 2024b) and OmniH2O (He et al., 2024a) utilize retargeted human motion priors (Mahmood et al., 2019). While foundation models like Octo (Team et al., 2024), GR00T (Bjorck et al., 2025), and RT-2 (Zitkovich et al., 2023) demonstrate that scaling data significantly boosts generalization, they primarily focus on manipulation or cross-embodiment alignment (Bohlinger et al., 2024; Huang et al., 2020). The foundation models of humanoid robots focus more on motion tracking (Zeng et al., 2025; Li et al., 2025). The closest work (Zhuang et al., 2023) employs distillation but struggles to generalize to novel composite terrains. Our framework bridges this gap by employing an autoregressive diffusion model trained on native kinematics, ensuring robust zero-shot adaptability without the limitations of retargeting or policy distillation.

2.2 Diffusion Models for Robot Learning

Beyond their success in vision (Song et al., 2020b; Ho et al., 2022), diffusion models have become an expressive generative paradigm in robot learning (Chi et al., 2023; Wang et al., 2023; Janner et al., 2022). Unlike traditional stepwise policies that directly map observations to actions, diffusion-based methods generate actions or trajectories by denoising from a prior, naturally supporting multi-modal behavior modeling and long-horizon temporal consistency (Wen et al., 2025; Liu et al., 2025). This property is particularly relevant for locomotion, where different terrains often require distinct but smoothly connected motion modes. DiPPeR (Liu et al., 2024a) utilizes diffusion for terrain-aware path planning, while DiffuseLoco (Huang et al., 2024) and BiRoDiff (Mothish et al., 2024) show that diffusion policies can unify diverse gait behaviors and adapt to unseen scenarios. Building on these advances, DreamPolicy uses a terrain-aware autoregressive diffusion planner to generate motion references conditioned on robot history and local terrain, guiding a unified reinforcement-learning controller across diverse terrains.

2.3 Motion Priors and Physics-Based Control

AMP (Peng et al., 2021) learns adversarial style rewards, while SMP (Mu et al., 2026) derives reusable rewards from pretrained diffusion models. CLoSD (Tevet et al., 2024) and ReactiveBFM (Chen et al., 2026) couple motion generation with execution feedback, while BeyondMimic (Liao et al., 2025) distills tracked skills into a guided diffusion policy. For terrain traversal, PARC (Xu et al., 2025) combines motion generation with physics-based data augmentation, and Perceptive Humanoid Parkour (Wu et al., 2026) uses motion matching and policy distillation to compose human skills. T-GMP (Guo et al., 2026) further introduces terrain-conditioned generative priors for adversarial motion learning. Building on these approaches, DreamPolicy learns from terrain-specialized robot rollouts and generates history and terrain-conditioned trajectories that jointly provide policy inputs, tracking targets, and adversarial motion references for unified locomotion.

3 Terrain-aware Humanoid Locomotion Skills Collection

In this section, we introduce our pipeline for collecting terrain-aware humanoid locomotion skill data. Section 3.1 outlines the modular terrain benchmarks designed to evaluate policy robustness. In section 3.2, we detail the training of specialized policies to obtain primitive skills.

3.1 Humanoid Locomotion Benchmarks with Diverse Terrains

We developed a modular humanoid terrain benchmark to facilitate skill acquisition and evaluation. This benchmark includes over a dozen specialized primitive terrains (e.g., Gap, Stair, Bridge) for

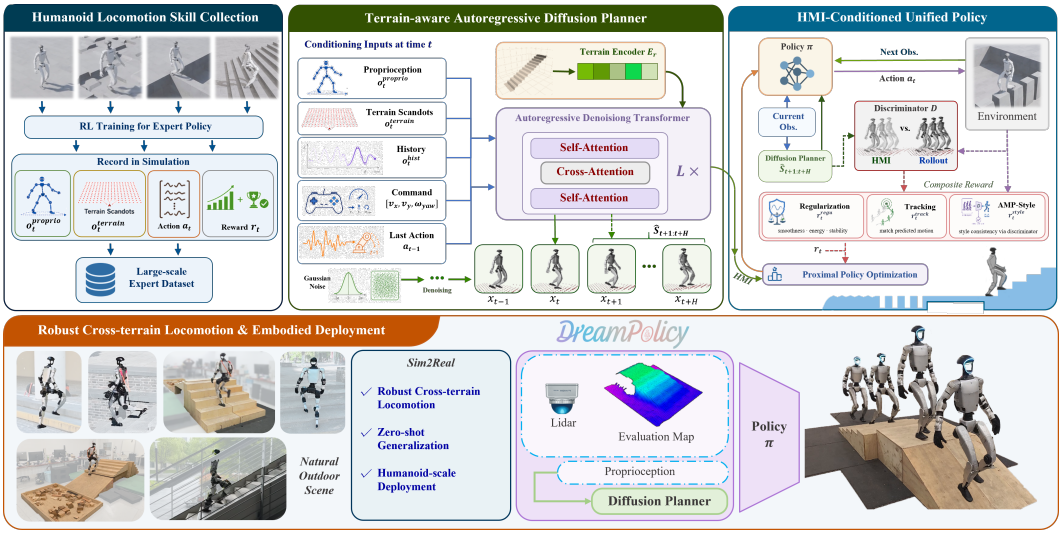


Figure 2: Framework of DreamPolicy. The system is decomposed into two parts: (1) Terrain-aware humanoid locomotion skill data collection, sufficient humanoid locomotion skill data on various challenging terrains will be collected; (2) DreamPolicy, training a unified policy across diverse terrains using skill data and humanoid motion imagery via an Auto-Regressive Diffusion Planner.

expert policy training, and challenging terrains (e.g., Balancing Beam, Slope-Bridge) for testing zero-shot generalization, as visualized in Figure 3. These terrains can be flexibly combined to generate a wide variety of scenarios, ensuring that the collected data covers a rich distribution of kinematic transitions. The benchmark supports both Isaac Gym and Isaac Lab, with aligned observation definitions, reward formulations, and terrain-generation protocols. Robot-specific configurations for the Unitree H1-2 and G1 further support benchmarking across embodiments. Details are provided in Appendix B.1.

Performance is measured by success rate and completion rate. An efficient, reusable recording module supports comprehensive data collection in simulation, capturing proprioception, exteroceptive data (e.g., sparse scandots), and rewards. Leveraging the simulator's parallelized architecture, we efficiently acquire terrain-specific policies. Ultimately, we collected data from five expert policies, yielding a dataset of 75 million transitions.

3.2 Humanoid Primitive Skills Collection via Specialized Policy Learning

To acquire primitive humanoid locomotion skills, we first train multiple physically feasible policies $\pi = [\pi_1, \pi_2, ..., \pi_n]$ in Isaaclab (Mittal et al., 2025) using reinforcement learning. Here, π_i represents an expert policy specialized for a single terrain.

While each policy is specialized for a single terrain and lacks generalization across composite scenarios (Section 5.1.2), they serve as high-quality primitives. Despite environmental variations, all policies share one network architecture for parameter sharing. Training follows a two-phase curriculum: a foundational locomotion policy is first trained from scratch under strict regularization for basic walking stability, then refined per terrain through terrain-specific reward shaping. This keeps reward engineering confined to the primitives, decoupled from generalized locomotion on novel terrains. Following (Zhu et al., 2026b), we incorporate edge-aware penalization to discourage risky footholds near terrain boundaries without requiring precise foothold specification.

Observation Space and Action Space. The policy obsevations, which denotes as o_t, consist of:

$$o_t = [o_t^{\text{proprio}}, o_t^{\text{terrain}}, o_t^{\text{hist}}, c_t, a_{t-1}], \tag{1}$$

where o_t^{proprio} is the proprioceptive state, o_t^{terrain} denotes exteroceptive information, o_t^{hist} represents the observation of the previous timestep, c_t is the user command, and a_{t-1} is the action of the last timestep. Proprioceptive information is defined as $o_t^{\text{proprio}} = [w_t, g_t, \theta_t, \dot{\theta}_t]$, containing base angular velocity, projected gravity vector, joint positions, and joint velocities. The command $c_t = [v_x, v_y, w_{\text{yaw}}]$

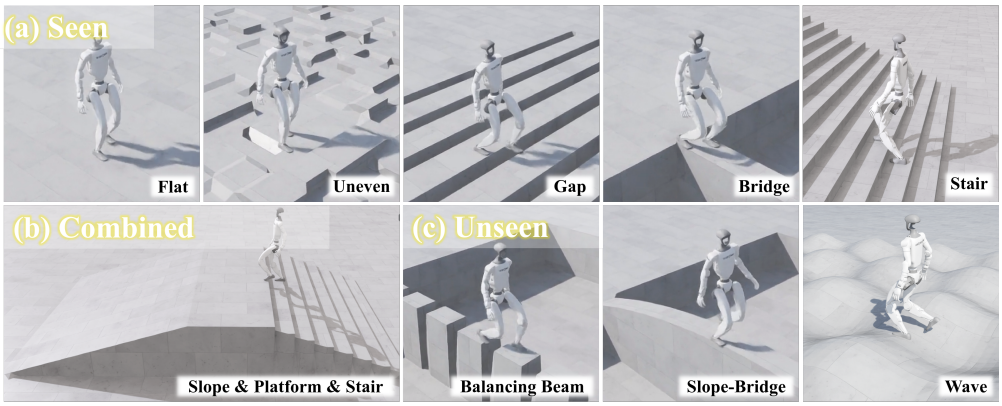


Figure 3: Visualization of the modular terrain benchmark. Top row: Primitive terrains used for skill collection, where multiple specialized expert policies are trained. Bottom row: Combined and unseen terrains for evaluating the zero-shot generalization of the unified policy.

includes the desired linear velocities and yaw rate, which are resampled randomly within a predefined range at fixed intervals during training to ensure robustness. Perceptive information o_t^{terrain} consists of sparse scandots sampled from the egocentric height map in the robot's frame, aligning its z-axis to the negative gravity direction. The action $a_t \in \mathbb{R}^{29}$ represents the target position for all joints of the robot. More details about the specialized policies are provided in the Appendix B.2.

4 Unified Policy for Cross-Terrain Humanoid Locomotion

In this section, we detail the implementation of the DreamPolicy framework, as illustrated in Figure 2. Section 4.1 introduces the Terrain-Aware Autoregressive Diffusion Planner that generates the HMI. Section 4.2 describes how we formulate the HMI-conditioned Unified Policy, which integrates HMI tracking with an AMP-style (Peng et al., 2021) manifold regularization.

4.1 Terrain-aware Autoregressive Diffusion Planner

Based on the large-scale multi-expert locomotion skill data collected in Section 3.2, we employ a terrain-aware autoregressive diffusion planner to predict adaptable humanoid future trajectories.

Autoregressive Denoising on State Space. To ensure the model is deployable on real robots without relying on privileged global ground truth, we define the humanoid state s_t consistently with the policy observation space defined in Section 3.2. Specifically, s_t is composed of the proprioceptive state, the command, and the previous action:

$$s_t = [o_t^{\text{proprio}}, c_t, a_{t-1}], \tag{2}$$

where the components follow the observation-space definitions in Section 3.2. Crucially, this state space is invariant to global position and yaw, enabling robust generalization across unbounded environments.

Rather than applying diffusion to actions (which are often discrete or vary in dimension), as analysed in (Ajay et al., 2023), we diffuse over the state trajectories. At each control step t, given the history of K frames of humanoid state $\mathbf{x}_{t-1} = [s_{t-K+1}, \dots, s_t]$ and current local terrain embedding $y_t = \mathcal{E}_{\text{terrain}}(o_t^{\text{terrain}})$, the goal is to sample the next H humanoid states $\mathbf{x}_t = [s_{t+1}, \dots, s_{t+H}]$ conditioned on (x_{t-1}, y_t).

For generative modeling, we follow a Denoising Diffusion Probabilistic Model (DDPM) (Ho et al., 2020) framework by gradually adding Gaussian noise to the clean humanoid state trajectory $\mathbf{x}_t^0$ over N steps. Formally, the forward process for diffusion timestep $n \in \{1, \dots, N\}$ is:

$$q(\mathbf{x}^n|\mathbf{x}^{n-1}) = \mathcal{N}(\sqrt{\alpha_n}\mathbf{x}^{n-1}, (1-\alpha_n)\mathbf{I}), \tag{3}$$

where $\alpha_n \in (0, 1)$ are fixed variance-scheduling parameter for step n. In the reverse process, we form the input to the denoiser as $(\mathbf{x}_t^n, \mathbf{x}_{t-1}, n, y_t)$. A transformer-based denoiser $\mathcal{G}$ is trained to predict

the cleaned humanoid state trajectory x_t^0 from the noisy input.

$$\hat{\mathbf{x}}_t^0 = \mathcal{G}(\mathbf{x}_t^n, \mathbf{x}_{t-1}, n, y_t), \tag{4}$$

where $\hat{x}_t^0$ is the model's estimate of the true states. Notably, we predict the states themselves rather than the noise terms, which can improve stability (Tevet et al., 2024). During training, terrain embedding is randomly masked by a rate of 0.1 for classifier-free guidance sampling at inference (Ho & Salimans, 2021).

Training. We train on our expert locomotion dataset by minimizing the mean-squared error between the predicted and true states. The diffusion reconstruction loss is:

$$\mathcal{L}_{\text{diffusion}} = \mathbb{E}_{\mathbf{x}_t^0 \sim q(\mathbf{x}_t^0), n \sim [1,N]}[\|\mathbf{x}_t^0 - \hat{\mathbf{x}}_t^0\|_2^2]. \tag{5}$$

During training, to improve stability and performance, we use a scheduled sampling strategy (Bengio et al., 2015; Yuan et al., 2023), where the state history is sampled at the probability p from the ground truth or from the model's own previous predictions. We gradually anneal the probability p of using true history from 1 to 0 (from fully teacher-forced to fully autoregressive). This curriculum aligns training and inference behavior, preventing divergence in long sequences.

Inference. After training a denoise network $\mathcal{G}$, we can auto-regressively generate the HMI given history states $\mathbf{x}_{t-1}$, a standard sampler $\mathcal{S}$ like (Ho et al., 2020; Song et al., 2020a) and online terrain embeddings y_t by sampling with classifier-free guidance (Ho & Salimans, 2021):

$$\begin{aligned} \mathcal{G}_\omega(\mathbf{x}_t^n, \mathbf{x}_{t-1}, n, y_t) = \mathcal{G}(\mathbf{x}_t^n, \mathbf{x}_{t-1}, n, \phi) \\ +\omega \cdot \Big(\mathcal{G}(\mathbf{x}_t^n, \mathbf{x}_{t-1}, n, y_t) - \mathcal{G}(\mathbf{x}_t^n, \mathbf{x}_{t-1}, n, \phi)\Big), \end{aligned} \tag{6}$$

where ω is the guidance hyper-parameter, detailed in Appendix B.3.

4.2 HMI-Conditioned Unified Policy

We formulate the unified policy learning as a Goal-Conditioned Reinforcement Learning (GCRL) problem (Liu et al., 2022). The goal g_t is the HMI $\hat{s}_{t+1:t+H}$ synthesized by the diffusion planner. To leverage the planner's predictive power, we design a hybrid reward combining precise kinematic tracking with motion manifold regularization.

Reward Design. Tracking only the immediate next frame underutilizes the model's ability to forecast coherent long-horizon motions, while style-only rewards yield ambiguous optimization objectives. Therefore, we propose a composite reward function:

$$r_t = w_{\text{regu}} r_t^{\text{regu}} + w_{\text{track}} r_t^{\text{track}} + w_{\text{style}} r_t^{\text{style}}, \tag{7}$$

where $w_{\text{regu}}, w_{\text{track}}, w_{\text{style}}$ are weighting coefficients.

Regularization Reward (r_t^{regu}): This term enforces fundamental physical constraints, including penalties for excessive joint positions, velocities, accelerations, and torques, as well as encouraging smooth base motion and contact stability.

Tracking Reward (r_t^{track}): Following the unified task-space formulation in BeyondMimic (Liao et al., 2025), we define an immediate body-tracking reward using the first predicted step $\hat{s}_{t+1}$ from the diffusion planner:

$$r_t^{\text{track}} = \sum_{s \in \{\mathbf{p}, R, \mathbf{v}, \boldsymbol{\omega}\}} r(\bar{e}_s, \sigma_s), \qquad r(\bar{e}_s, \sigma_s) = \exp\left(-\bar{e}_s / \sigma_s^2\right). \tag{8}$$

where the sum runs over the four tracked body-space quantities $s \in \{\mathbf{p}, R, \mathbf{v}, \boldsymbol{\omega}\}$, i.e., position, rotation, and linear and angular velocity of the bodies in $\mathcal{B}_{\text{target}}$, which are represented in a body-relative frame, and $\bar{e}_s$ denotes the mean tracking error of term s with tolerance σ_s.

Style Reward (r_t^{style}): The style reward r_t^{style} is computed using an AMP (Peng et al., 2021) approach adapted for online diffusion generation (Algorithm 3). A discriminator D is jointly trained to

distinguish between the policy's actual transition (s_t, s_{t+1}) and the "generated" transition $(\hat{s}_t, \hat{s}_{t+1})$ predicted by the diffusion planner. The objective of our discriminator is formulated as:

$$\arg\min_D \mathbb{E}_{(\hat{s}_t,\hat{s}_{t+1})\sim p_{\text{diff}}}\left[(D(\hat{s}_t,\hat{s}_{t+1})-1)^2\right] + \mathbb{E}_{(s_t,s_{t+1})\sim\pi}\left[(D(s_t,s_{t+1})+1)^2\right], \tag{9}$$

where $\hat{s}_t$, $\hat{s}_{t+1}$ are state transitions sampled from the diffusion planner, s_t, s_{t+1} are the policy rollouts. The style reward is then defined as:

$$r_t^{\text{style}} = \max\left\{0, 1 - 0.25(D(\Phi(s_t),\Phi(s_{t+1}))-1)^2\right\}, \tag{10}$$

where Φ denotes a fixed state feature mapping applied to each state before it is passed to the discriminator. This reward encourages the policy to produce trajectories that align with the diffusion planner's implicit knowledge of stable, adaptive locomotion over the prediction horizon. The standard AMP (Peng et al., 2021) relies on offline-sampled action sequences from fixed datasets; our method uniquely employs online inference conditioned on the robot's real-time state $s_{t-K+1:t}$ to generate the sequence set $\hat{s}_{t+1:t+H}$. Therefore, a more powerful discriminator is trained on real-time policy trajectories and diffusion-filtered style samples.

Overall, the unified policy uses 19 scalar reward terms: 4 body-space tracking terms, 10 physical regularization terms, 4 safety terms, and 1 online diffusion-AMP style term. Excluding the shared embodiment-level regularization and safety terms, only 5 terms are non-shared, namely 4 diffusion-guided tracking terms and 1 style term, against 14 non-shared terms in TTT-Parkour (Zhu et al., 2026a). Thus, terrain-specific shaping is amortized into the expert stage, paid once at data-collection time rather than re-tuned for every new terrain combination. More details are provided in Appendix B.4.

5 Experiment

We evaluate DreamPolicy across complex terrain scenarios: first the humanoid locomotion benchmarks and comparisons against prior work, then ablations and scaling with data, and finally sim-to-real transfer on hardware.

5.1 Simulation Experiments

5.1.1 Experimental Setup

Comparison Methods. We compare our method against the following settings: 1) scenario specialized policy that extends the Actor-Critic architecture from (Rudin et al., 2022) with a scandot encoder. 2) distillation-based method following DAgger (Chen et al., 2020; Ross et al., 2011); 3) a joint multi-terrain policy trained with the same configuration as the scenario specialized policy, but on our final mixed terrain; 4) ablation without the Diffusion Planner; 5) ablation without Goal-Conditioned RL. Design details are provided in Appendix C.1.

Scenarios. We evaluated our method on 1) **Five single-terrains** *(Flat, Gap, Stair, Bridge, Uneven)*. 2) **Four unseen terrains** *(Wave, Slope , Balancing Beam, Slope bridge)*. As shown in Figure 3. Design details are provided in Appendix B.1.

Setup. Simulation results use Isaac Lab (Mittal et al., 2025) with the Unitree G1. Each specialized policy is trained with PPO (Schulman et al., 2017) in 4096 parallel environments, and the offline expert set aggregates five terrains, three episodes per terrain over six rounds, giving $\approx$75M frames of transitions. Terrain geometry and curriculum parameters follow Appendix B.1.

Metrics. We evaluate all settings with the following metrics. 1) Success Rate R_{succ}: The percentage of successful attempts to cross the entire terrain. 2) Complete Rate R_{cmp}: The ratio of the distance traveled before falling to the total terrain length. All simulation results average five seeds of $40{,}960$ episodes each, and both rates come from those outcomes.

5.1.2 Result

Single-terrain. DreamPolicy matches expert performance without the distillation baselines' regression (Fig. A3, Table A9), effectively learning a unified locomotion policy. **Unseen terrains.** For zero-shot generalization on four novel terrains (Table 1), our method demonstrates superior



Figure 4: **Real-world experiment.** We test our policy on real-world stairs, platforms, slopes and uneven terrain, similar to those in simulation.

Table 1: **Comprehensive results across unseen terrain types.** DreamPolicy against four baselines on four unseen terrains, under Success Rate and Complete Rate; higher is better.

	Success Rate (%) ↑				Complete Rate (%) ↑			
Method	**Slope bridge**	**Balancing Beam**	**Wave**	**Slope**	**Slope bridge**	**Balancing Beam**	**Wave**	**Slope**
Ours	**92.77**	**96.35**	**99.59**	**99.54**	**96.80**	**98.33**	99.72	**99.70**
Distillation	65.48	74.36	99.41	99.37	69.54	78.43	**99.76**	99.66
Joint multi-terrain Policy	31.24	41.36	99.40	99.12	37.96	48.73	99.70	99.55
w/o Diffusion Planner	0.08	20.14	97.32	98.65	10.23	35.79	98.37	99.13
w/o GCRL	0.05	0.12	60.38	94.35	1.39	5.43	80.21	97.37

performance over prior approaches. **Combined terrains.** Under compounding complexity (Table 2), DreamPolicy significantly outperforms baselines. The gap is notably striking on challenging combinations: yielding 37.98% and 22.45% improvements over the baseline on 'Stair&Uneven' and 'Bridge&Uneven', while maintaining over 91% success elsewhere, showing strong generalization and robustness on complex terrain (Appendix C.3).

5.1.3 Ablation Analysis

To isolate each component's contribution, we ablate the Diffusion Planner and Goal-Conditioned RL.

Ours w./w.o. Diffusion Planner. Removing the diffusion planner forces the agent to rely solely on the aggregated offline expert data. Performance then declines sharply (Table 1) and worsens rapidly with terrain complexity. Without future trajectories to follow, the policy fails to generalize to out-of-distribution (OOD) states and cannot adapt to unseen complex environments.

Ours w./w.o. GCRL. Removing the goal-conditioned tracker lets the diffusion model control the robot end-to-end and emit joint targets rather than states, which amounts to a direct-action diffusion policy. It performs well on simple terrains (Figure A3) but fails on more complex environments (Table 1). While the diffusion planner can generate optimal policies, these are not always physically feasible and can be disrupted by environmental factors such as stumbling on uneven ground.

5.1.4 Scaling Analysis

DreamPolicy gains consistently as the offline dataset grows from 15M to 75M transitions (Fig. 5a) and from 1 to 5 terrain types (Fig. 5b), with far higher data efficiency than standard distillation.

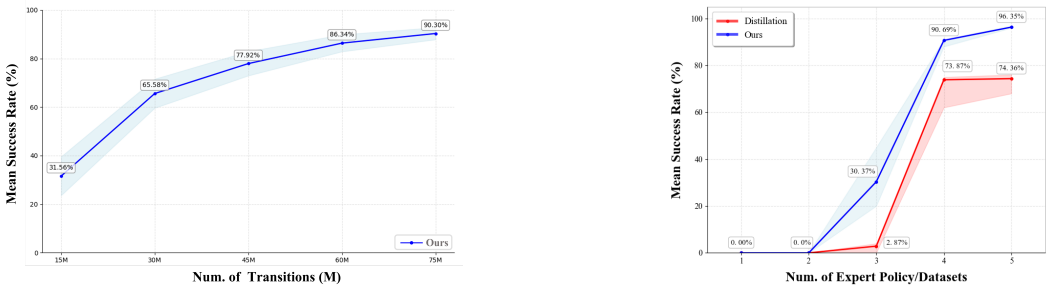
5.2 Real-world Experiments

Real-world setting. The Unitree G1 runs a distributed ROS stack: the onboard computer performs perception with FAST-LIO2 (Xu et al., 2022) and elevation mapping (Fankhauser et al., 2018) into the robot-centric height grid used in training ($\approx$20 Hz), while an external PC runs inference and an asynchronous buffer decouples the 50 Hz controller from this perception rate and the inference latency (Appendix C.4). We deploy the policies zero-shot, without real-world fine-tuning: the robot traverses stairs, slopes, gaps, uneven surfaces, and their combinations, crossing discrete gaps and climbing stairs continuously without pausing (Fig. 4).

Quantitative analysis. Table 3 reports 20 trials per method on each of four real-world terrains. DreamPolicy succeeds in 19/20, 17/20, 16/20 and 17/20 trials on Slope, Stair, Gap and Balancing Beam, against 19/20, 16/20, 8/20 and 9/20 for the distillation baseline, while the jointly trained multi-terrain policy does not exceed 17/20 on any terrain. The gain is thus concentrated where the terrain

Table 2: **Performance Comparison in combined terrain.** All values are percentages; the higher the better, best in boldface.

	Stair&Uneven		Bridge&Uneven		Stair&Gap		Stair&Bridge		Bridge&Gap	
	$R_{\text{succ}}(\%)$	$R_{\text{cmp}}(\%)$	$R_{\text{succ}}(\%)$	$R_{\text{cmp}}(\%)$	$R_{\text{succ}}(\%)$	$R_{\text{cmp}}(\%)$	$R_{\text{succ}}(\%)$	$R_{\text{cmp}}(\%)$	$R_{\text{succ}}(\%)$	$R_{\text{cmp}}(\%)$
Distillation	50.48	76.40	62.33	74.49	88.42	90.28	86.76	90.66	91.34	95.26
Joint multi-terrain Policy	42.15	60.04	54.62	68.09	78.35	85.37	75.46	84.04	86.52	94.95
Ours	**88.46**	**96.23**	**84.78**	**90.73**	**95.79**	**98.78**	**91.45**	**95.60**	**94.68**	**96.67**



(a) Scaling law of numbers

(b) Scaling law of types

Figure 5: **Scalability Analysis for dataset size and types.** Both subfigures demonstrate a consistent increase in the mean success rate as the offline dataset scales up in either (a) size or (b) diversity. Success rates are measured by rolling out the final policy in the combined-terrain simulation environments (§5.1.1); shaded regions denote the standard deviation across seeds.

Table 3: **Quantitative real-world evaluation.** Success rate over 20 trials per method and terrain; a trial succeeds if the robot reaches the terminal platform without falling or requiring intervention.

Method	**Slope**	**Stair**	**Gap**	**Balancing Beam**
Distillation	**19/20 (95%)**	16/20 (80%)	8/20 (40%)	9/20 (45%)
Joint multi-terrain Policy	17/20 (85%)	11/20 (55%)	6/20 (30%)	7/20 (35%)
Ours	**19/20 (95%)**	**17/20 (85%)**	**16/20 (80%)**	**17/20 (85%)**

demands accurate foot placement: on Gap and Balancing Beam the 40-point margin over distillation (50 points over the joint policy) is significant under a two-sided Fisher exact test ($p < 0.05$), whereas on the two continuous surfaces the methods are statistically indistinguishable (19/20 each on Slope; 17 versus 16 on Stair). The learned motion prior therefore pays off on discrete, spatially demanding terrain rather than uniformly, mirroring simulation, where the margin over distillation is likewise under 0.5 points on the near-saturated Wave and Slope terrains but 22–27 points on Slope bridge and Balancing Beam (Table 1).

6 Conclusion, Limitations and Future Works

We present DreamPolicy, a scalable framework for humanoid locomotion through terrain-aware generative planning. Built on a modular multi-terrain benchmark and large-scale offline experience from specialized policies, it learns a terrain-aware autoregressive diffusion planner that generates physically plausible future state trajectories conditioned on the robot's state history and terrain information, and a unified HMI-conditioned policy that tracks them, allowing locomotion behaviors to be dynamically adapted to the encountered terrain without terrain-specific reward design at deployment. Separating terrain-aware motion generation from low-level control lets DreamPolicy exploit growing offline experience and generalize beyond the terrain distributions of its source policies. Results on unseen and composite terrains, together with zero-shot transfer to a real Unitree G1, demonstrate the potential of generative planning as a scalable paradigm for generalist humanoid locomotion. Its remaining limitation is the iterative denoising overhead; future work will explore lightweight sampling strategies to accelerate inference for synchronous, high-frequency planning without sacrificing trajectory quality.

AI USE STATEMENT

We did not use generative AI tools for any task that ICLR requires to be disclosed, including dataset generation, method implementation, experimental design, and interpretation of results. Generative AI tools were used for recommended-disclosure tasks: editing the manuscript to improve readability, i.e. polishing the wording and grammar of text the authors had already written, and modifying some of the figures, whose quantitative content was specified and verified by the authors. All AI-assisted material was reviewed by the authors, who take responsibility for the final content of this work.

ETHICS STATEMENT

We adhere to the ICLR Code of Ethics. This work involves simulated locomotion experiments and in-lab robot experiments conducted under standard safety protocols, including emergency-stop procedures, safety clearance zones, and supervised operation by trained personnel. No human subjects, personally identifiable information, or sensitive biometric data were collected, and no third-party datasets were used. The intended application is general-purpose legged locomotion over rough terrain; as with any locomotion controller deployed on hardware, the physical safety of nearby people and of the platform itself is a deployment-time concern, and the failure cases analysed in the appendix are reported to make the current limitations of the approach explicit rather than to overstate its robustness.

REPRODUCIBILITY STATEMENT

The main paper describes the proposed method and reports the principal experimental results. The appendix provides further information on the terrain benchmarks, the specialized policy training protocol, the architecture and training hyperparameters of the diffusion planner, the design of the unified policy, and the implementation of the baselines used for comparison, so as to support reproduction. We plan to release relevant code and supporting materials when possible, subject to the applicable licenses and access restrictions.

REFERENCES

Anurag Ajay, Yilun Du, Abhi Gupta, Joshua B. Tenenbaum, Tommi S. Jaakkola, and Pulkit Agrawal. Is conditional generative modeling all you need for decision making? In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=sP1fo2K9DFG.

Mert Albaba, Chenhao Li, Markos Diomataris, Omid Taheri, Andreas Krause, and Michael Black. Nil: No-data imitation learning by leveraging pre-trained video diffusion models. *arXiv preprint arXiv:2503.10626*, 2025.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. *Advances in neural information processing systems*, 28, 2015.

Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

Nico Bohlinger, Grzegorz Czechmanowski, Maciej Krupka, Piotr Kicki, Krzysztof Walas, Jan Peters, and Davide Tateo. One policy to run them all: an end-to-end learning approach to multi-embodiment locomotion. *Conference on Robot Learning*, 2024.

Dian Chen, Brady Zhou, Vladlen Koltun, and Philipp Krähenbühl. Learning by cheating. In *Conference on robot learning*, pp. 66–75. PMLR, 2020.

Jingxiao Chen, Xinyao Li, Jiahang Cao, Zhengbang Zhu, Wentao Dong, Minghuan Liu, Ying Wen, Yong Yu, Liqing Zhang, and Weinan Zhang. Rhino: Learning real-time humanoid-human-object interaction from human demonstrations. *arXiv preprint arXiv:2502.13134*, 2025.

Xiao Chen, Weishuai Zeng, Xiaojie Niu, Zirui Wang, Jianan Li, Huayi Wang, Furui Xu, Jiahe Chen, Weixiang Zhong, Lihe Ding, Kailin Li, Jiangmiao Pang, Tai Wang, Tianfan Xue, and Jingbo Wang. Reactivebfm: Reactive closed-loop motion planning towards universal humanoid whole-body control, 2026. URL https://arxiv.org/abs/2606.30362.

Xuxin Cheng, Yandong Ji, Junming Chen, Ruihan Yang, Ge Yang, and Xiaolong Wang. Expressive whole-body control for humanoid robots. *arXiv preprint arXiv:2402.16796*, 2024a.

Xuxin Cheng, Jialong Li, Shiqi Yang, Ge Yang, and Xiaolong Wang. Open-television: Teleoperation with immersive active visual feedback. *arXiv preprint arXiv:2407.01512*, 2024b.

Xuxin Cheng, Kexin Shi, Ananye Agarwal, and Deepak Pathak. Extreme parkour with legged robots. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 11443–11450. IEEE, 2024c.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, pp. 02783649241273668, 2023.

Péter Fankhauser, Michael Bloesch, and Marco Hutter. Probabilistic terrain mapping for mobile robots with uncertain localization. *IEEE Robotics and Automation Letters*, 3(4):3019–3026, 2018.

Zipeng Fu, Ashish Kumar, Jitendra Malik, and Deepak Pathak. Minimizing energy consumption leads to the emergence of gaits in legged robots. In *Conference on Robot Learning (CoRL)*, 2021.

Zipeng Fu, Xuxin Cheng, and Deepak Pathak. Deep whole-body control: Learning a unified policy for manipulation and locomotion. In *Conference on Robot Learning (CoRL)*, 2022.

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

Xinyang Gu, Yen-Jen Wang, Xiang Zhu, Chengming Shi, Yanjiang Guo, Yichen Liu, and Jianyu Chen. Advancing humanoid locomotion: Mastering challenging terrains with denoising world model learning. *arXiv preprint arXiv:2408.14472*, 2024.

Zhaoyuan Gu, Junheng Li, Wenlan Shen, Wenhao Yu, Zhaoming Xie, Stephen McCrory, Xianyi Cheng, Abdulaziz Shamsah, Robert Griffin, C Karen Liu, et al. Humanoid locomotion and manipulation: Current progress and challenges in control, planning, and learning. *arXiv preprint arXiv:2501.02116*, 2025.

Junhong Guo, Hao Hu, Chen Chen, Haoxuan Han, Linao Gong, Xin Yang, Zhicheng He, Yao Su, and Fenghua He. T-gmp: Terrain-conditioned generative motion priors for versatile and natural humanoid locomotion, 2026. URL https://arxiv.org/abs/2606.06944.

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. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 8944–8951. IEEE, 2024b.

Tairan He, Wenli Xiao, Toru Lin, Zhengyi Luo, Zhenjia Xu, Zhenyu Jiang, Changliu Liu, Guanya Shi, Xiaolong Wang, Linxi Fan, and Yuke Zhu. Hover: Versatile neural whole-body controller for humanoid robots. *arXiv preprint arXiv:2410.21229*, 2024c.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in Neural Information Processing Systems*, 35:8633–8646, 2022.

Wenlong Huang, Igor Mordatch, and Deepak Pathak. One policy to control them all: Shared modular policies for agent-agnostic control. In *International Conference on Machine Learning*, pp. 4455–4464. PMLR, 2020.

Xiaoyu Huang, Yufeng Chi, Ruofeng Wang, Zhongyu Li, Xue Bin Peng, Sophia Shao, Borivoje Nikolic, and Koushil Sreenath. DiffuseLoco: Real-time legged locomotion control with diffusion from offline datasets. In *8th Annual Conference on Robot Learning*, 2024.

Michael Janner, Yilun Du, Joshua Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. In *International Conference on Machine Learning*, pp. 9902–9915. PMLR, 2022.

Fabian Jenelten, Jemin Hwangbo, Fabian Tresoldi, C Dario Bellicoso, and Marco Hutter. Dynamic locomotion on slippery ground. *IEEE Robotics and Automation Letters*, 4(4):4170–4176, 2019.

Zhenyu Jiang, Yuqi Xie, Jinhan Li, Ye Yuan, Yifeng Zhu, and Yuke Zhu. Harmon: Whole-body motion generation of humanoid robots from language descriptions. *arXiv preprint arXiv:2410.12773*, 2024.

Yitang Li, Zhengyi Luo, Tonghe Zhang, Cunxi Dai, Anssi Kanervisto, Andrea Tirinzoni, Haoyang Weng, Kris Kitani, Mateusz Guzek, Ahmed Touati, et al. Bfm-zero: A promptable behavioral foundation model for humanoid control using unsupervised reinforcement learning. *arXiv preprint arXiv:2511.04131*, 2025.

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.

Jiaming Liu, Hao Chen, Pengju An, Zhuoyang Liu, Renrui Zhang, Chenyang Gu, Xiaoqi Li, Ziyu Guo, Sixiang Chen, Mengzhen Liu, et al. Hybridvla: Collaborative diffusion and autoregression in a unified vision-language-action model. *arXiv preprint arXiv:2503.10631*, 2025.

Jianwei Liu, Maria Stamatopoulou, and Dimitrios Kanoulas. Dipper: Diffusion-based 2d path planner applied on legged robots. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 9264–9270. IEEE, 2024a.

Minghuan Liu, Menghui Zhu, and Weinan Zhang. Goal-conditioned reinforcement learning: Problems and solutions. *arXiv preprint arXiv:2201.08299*, 2022.

Minghuan Liu, Zixuan Chen, Xuxin Cheng, Yandong Ji, Rizhao Qiu, Ruihan Yang, and Xiaolong Wang. Visual whole-body control for legged loco-manipulation. *The 8th Conference on Robot Learning*, 2024b.

Junfeng Long, Junli Ren, Moji Shi, Zirui Wang, Tao Huang, Ping Luo, and Jiangmiao Pang. Learning humanoid locomotion with perceptive internal model, 2024. URL https://arxiv.org/abs/2411.14386.

Haofei Lu, Dongqi Han, Yifei Shen, and Dongsheng Li. What makes a good diffusion planner for decision making? In *International conference on learning representations*, volume 2025, pp. 68505–68534, 2025.

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*, pp. 5442–5451, 2019.

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

GVS Mothish, Manan Tayal, and Shishir Kolathaya. Birodiff: Diffusion policies for bipedal robot locomotion on unseen terrains. *arXiv preprint arXiv:2407.05424*, 2024.

Yuxuan Mu, Ziyu Zhang, Yi Shi, Dun Yang, Minami Matsumoto, Kotaro Imamura, Guy Tevet, Chuan Guo, Michael Taylor, Chang Shu, Pengcheng Xi, and Xue Bin Peng. Smp: Reusable score-matching motion priors for physics-based character control, 2026. URL https://arxiv.org/abs/2512.03028.

Xue Bin Peng, Ze Ma, Pieter Abbeel, Sergey Levine, and Angjoo Kanazawa. Amp: Adversarial motion priors for stylized physics-based character control. *ACM Transactions on Graphics (ToG)*, 40(4):1–20, 2021.

Ilija Radosavovic, Sarthak Kamat, Trevor Darrell, and Jitendra Malik. Learning humanoid locomotion over challenging terrain. *arXiv preprint arXiv:2410.03654*, 2024a.

Ilija Radosavovic, Bike Zhang, Baifeng Shi, Jathushan Rajasegaran, Sarthak Kamat, Trevor Darrell, Koushil Sreenath, and Jitendra Malik. Humanoid locomotion as next token prediction. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024b.

Junli Ren, Tao Huang, Huayi Wang, Zirui Wang, Qingwei Ben, Jiangmiao Pang, and Ping Luo. Vb-com: Learning vision-blind composite humanoid locomotion against deficient perception. *arXiv preprint arXiv:2502.14814*, 2025.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

Nikita Rudin, David Hoeller, Philipp Reist, and Marco Hutter. Learning to walk in minutes using massively parallel deep reinforcement learning. In *Conference on Robot Learning*, pp. 91–100. PMLR, 2022.

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

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020a.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020b.

Wandong Sun, Baoshi Cao, Long Chen, Yongbo Su, Yang Liu, Zongwu Xie, and Hong Liu. Learning perceptive humanoid locomotion over challenging terrain. *arXiv preprint arXiv:2503.00692*, 2025.

Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, et al. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*, 2024.

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. *arXiv preprint arXiv:2410.03441*, 2024.

Miomir Vukobratović and Juri Stepanenko. On the stability of anthropomorphic systems. *Mathematical biosciences*, 15(1-2):1–37, 1972.

Dewei Wang, Xinmiao Wang, Chenyun Zhang, Jiyuan Shi, Yingnan Zhao, Chenjia Bai, and Xuelong Li. X-loco: Towards generalist humanoid locomotion control via synergetic policy distillation. *arXiv preprint arXiv:2603.03733*, 2026.

Huayi Wang, Zirui Wang, Junli Ren, Qingwei Ben, Tao Huang, Weinan Zhang, and Jiangmiao Pang. BeamDojo: Learning agile humanoid locomotion on sparse footholds. In *Robotics: Science and Systems (RSS)*, 2025.

Zhendong Wang, Jonathan J Hunt, and Mingyuan Zhou. Diffusion policies as an expressive policy class for offline reinforcement learning. In *The Eleventh International Conference on Learning Representations*, 2023.

Junjie Wen, Yichen Zhu, Minjie Zhu, Zhibin Tang, Jinming Li, Zhongyi Zhou, Xiaoyu Liu, Chaomin Shen, Yaxin Peng, and Feifei Feng. Diffusionvla: Scaling robot foundation models via unified diffusion and autoregression. In *Forty-second International Conference on Machine Learning*, 2025.

Zhen Wu, Xiaoyu Huang, Lujie Yang, Yuanhang Zhang, Koushil Sreenath, Xi Chen, Pieter Abbeel, Rocky Duan, Angjoo Kanazawa, Carmelo Sferrazza, et al. Perceptive humanoid parkour: Chaining dynamic human skills via motion matching. *arXiv preprint arXiv:2602.15827*, 2026.

Weiji Xie, Chenjia Bai, Jiyuan Shi, Junkai Yang, Yunfei Ge, Weinan Zhang, and Xuelong Li. Humanoid whole-body locomotion on narrow terrain via dynamic balance and reinforcement learning. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4751–4758. IEEE, 2025.

Michael Xu, Yi Shi, KangKang Yin, and Xue Bin Peng. Parc: Physics-based augmentation with reinforcement learning for character controllers. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pp. 1–11, 2025.

Wei Xu, Yixi Cai, Dongjiao He, Jiarong Lin, and Fu Zhang. Fast-lio2: Fast direct lidar-inertial odometry. *IEEE Transactions on Robotics*, 38(4):2053–2073, 2022.

Yufei Xue, Wentao Dong, Minghuan Liu, Weinan Zhang, and Jiangmiao Pang. A unified and general humanoid whole-body controller for fine-grained locomotion. In *Robotics: Science and Systems (RSS)*, 2025.

Ye Yuan, Viktor Makoviychuk, Y Guo, S Fidler, XB Peng, and K Fatahalian. Learning physically simulated tennis skills from broadcast videos. *ACM Trans. Graph*, 42(4), 2023.

Fatemeh Zargarbashi, Jin Cheng, Dongho Kang, Robert Sumner, and Stelian Coros. Robotkeyframing: Learning locomotion with high-level objectives via mixture of dense and sparse rewards. *arXiv preprint arXiv:2407.11562*, 2024.

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

Qiang Zhang, Peter Cui, David Yan, Jingkai Sun, Yiqun Duan, Gang Han, Wen Zhao, Weining Zhang, Yijie Guo, Arthur Zhang, et al. Whole-body humanoid robot locomotion with human reference. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 11225–11231. IEEE, 2024.

Qiang Zhang, Gang Han, Jingkai Sun, Wen Zhao, Chenghao Sun, Jiahang Cao, Jiaxu Wang, Yijie Guo, and Renjing Xu. Distillation-ppo: A novel two-stage reinforcement learning framework for humanoid robot perceptive locomotion. *arXiv preprint arXiv:2503.08299*, 2025a.

Xianqi Zhang, Hongliang Wei, Wenrui Wang, Xingtao Wang, Xiaopeng Fan, and Debin Zhao. Flam: Foundation model-based body stabilization for humanoid locomotion and manipulation. *arXiv preprint arXiv:2503.22249*, 2025b.

Kaifeng Zhao, Gen Li, and Siyu Tang. Dartcontrol: A diffusion-based autoregressive motion model for real-time text-driven motion control. In *The Thirteenth International Conference on Learning Representations*, 2024.

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

Shaoting Zhu, Ziwen Zhuang, Mengjie Zhao, Kun-Ying Lee, and Hang Zhao. Hiking in the wild: A scalable perceptive parkour framework for humanoids. *arXiv preprint arXiv:2601.07718*, 2026b.

Ziwen Zhuang, Zipeng Fu, Jianren Wang, Christopher Atkeson, Soeren Schwertfeger, Chelsea Finn, and Hang Zhao. Robot parkour learning. *arXiv preprint arXiv:2309.05665*, 2023.

Ziwen Zhuang, Shenzhe Yao, and Hang Zhao. Humanoid parkour learning. In *8th Annual Conference on Robot Learning*, 2024. URL https://openreview.net/forum?id=fs7ia3FqUM.

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

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pp. 2165–2183. PMLR, 2023.

A Overview

The appendix provides comprehensive supplementary materials supporting the methodological and empirical contributions of this work. Section B elaborates on technical implementations, including the configuration of humanoid locomotion benchmarks (B.1), specialized policy training protocols (B.2), architectural details of the autoregressive diffusion planner (B.3), and the design principles of the diffusion guided unified policy (B.4). Section C presents extended experimental analyses, encompassing baseline implementation specifics (C.1), additional performance evaluations across diverse scenarios (C.2), qualitative visualizations (C.3), and supplementary video demonstrations (D). Section E concludes with an in-depth discussion of limitations through failure case examinations (E.1) and potential future research directions (E.3).

B Supplementary Technical Details

B.1 Details of Humanoid Locomotion Benchmarks

Curriculum design For the parameters in the table, $p[0]$ represents the first dimension, $p[1]$ represents the second dimension. If $p[0] > p[1]$, the larger the parameter, the simpler it is. The curriculum design is as follows:

$$\text{Truth parameters} = \text{random}\left(\frac{p[1]-p[0]}{\text{max_terrain_level}} \times \text{Difficulty}\right) + p[0] \tag{11}$$

Detailed parameters of different terrains. For all terrains, we add base roughness to make the ground undulating. The height of the ground is between $\text{random}(-0.02, 0.02) \times \text{Difficulty}$. The difficulty increases with terrain level. Notably, prior research typically used test terrains longer than 4m (Xie et al., 2025) and 8m (Wang et al., 2025) for single terrains. To ensure a more robust evaluation and minimize random effects, we increase the test terrain length to 10m.

The five Single-Terrains' specific parameters are shown in Table A1. For the *Gap*, the Size Range represents the platform that robots can stand, the gap robots need to cross is $[0.2, 0.3]$.

Table A1: **Single-Terrain curriculum parameter.** We designed different curriculum parameters for different Single-Terrains.

	Parameter	*Height_range (m)*	*Size_range (m)*	*Subsection_nums*
Single Terrain	Flat	–	–	–
	Uneven	$[0.10, 0.20]$	$[0.40, 0.70]$	$[150, 200]$
	Stair	$[0.15, 0.30]$	$[0.40, 0.25]$	$[5, 10]$
	Gap	–	$[0.60, 0.40]$	$[5, 10]$
	Bridge	–	$[0.40, 0.20]$	–

Due to the particularity of the four **Unseen-Terrains**, we will introduce each terrain separately.

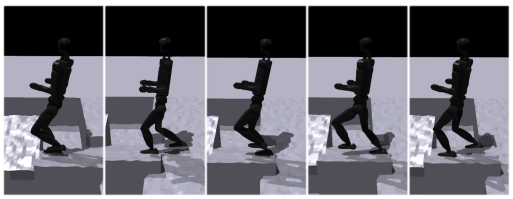
- The *Wave* adds several sinusoidal functions to make the ground undulate. The number of sine functions $[5, 10]$, and the amplitude of each sine function $[0.2, 0.5]$. This terrain is designed to test whether the robot can maintain a stable center of gravity when facing disturbances from irregular ground.
- The *Slope* had a fixed inclination angle of 30°. The surface was covered with a coarse-grit sandpaper to provide a consistent, high-friction interface with a measured coefficient of static friction.
- In *Balancing Beam*, the width of the standing platform across the gap is set to match the bridge. This configuration is designed to evaluate how well the robot integrates two distinct locomotion strategies.
- The *Slope-Bridge* is a bridge whose height is set according to the first half of the sine function. The slope of an arch bridge is determined by the length and amplitude of the terrain. Since we have fixed the length of the terrain to 18 meters, the amplitude is set to

$[1.5, 2.5]$ meters, with the width as a single bridge. This terrain is designed to test the robot's ability to generalize from stair terrain to slopes.

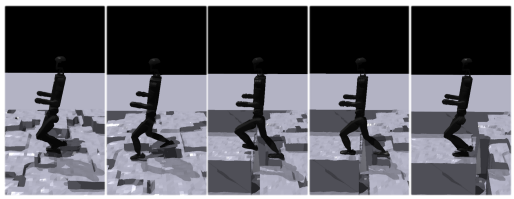
Cross embodiment and Cross simulation To support the research community's transition toward next-generation simulation standards, we provide distinct, fully validated implementations of our framework for both Isaac Gym and Isaac Lab (NVIDIA Omniverse). While the Isaac Gym version offers a highly optimized backend for rapid, high-throughput training on consumer hardware, the Isaac Lab implementation leverages the advanced PhysX 5 engine and high-fidelity rendering to enhance simulation accuracy and Sim-to-Real transfer. Both implementations adhere to a strictly aligned benchmark with identical observation spaces, reward formulations, and terrain generation to ensure algorithmic training across different simulation platforms.

We demonstrate the cross-embodiment scalability of DreamPolicy by validating it on distinct robots, including the Unitree H1-2 and Unitree G1. The inclusion of specialized configuration presets for these platforms confirms that our approach generalizes effectively across dynamic systems without overfitting to specific kinematics. Furthermore, the modular design of our reward and observation structures allows researchers to seamlessly integrate new URDF/MJCF models, facilitating the rapid benchmarking of novel hardware platforms within our standardized environments.

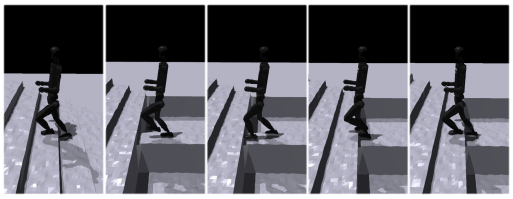
More visualization results are shown in Fig. A1, and Figure A2.



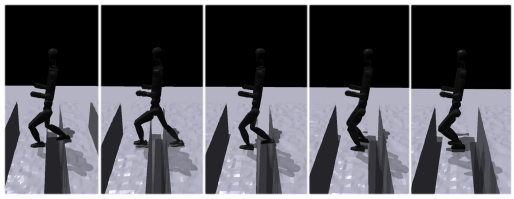
(a) Simulation qualitative result on parkour terrain.



(b) Simulation qualitative result on uneven terrain.



(c) Simulation qualitative result on bridge terrain.



(d) Simulation qualitative result on gap.

Figure A1: Simulation qualitative result in IsaacGym.

Detailed expert data collection. We collected expert data from three episodes in five terrains, six rounds each, and 1024 environments, totaling about 75M frames of data. In each frame, we record the following information, as shown in the Table A2.

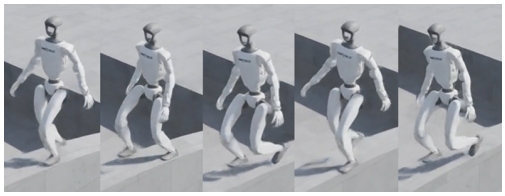
Table A2: Robot Locomotion Expert Data Items.

Terrain Scandot	(T, 187)
Robot Observations	(T, 98)
Actions	(T, 29)

Domain randomization. During training we apply random push perturbations to the robot base as part of the standard domain randomization. This broadens the state distribution seen by the tracking policy, so that after a deviation it can still recover whenever the re-planned reference remains reachable.



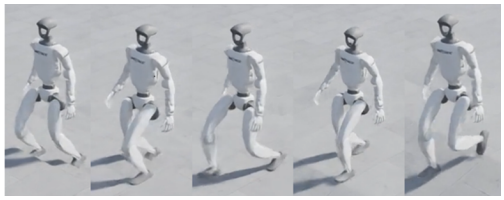
(a) Simulation qualitative Results on Balancing Beam.



(b) Simulation qualitative result on Slope-Bridge.



(c) Simulation qualitative result on Wave Terrain.



(d) Simulation qualitative result on multi-terrain.

Figure A2: Simulation qualitative result in IsaacLab.

B.2 Details of Specialized Policy Learning

Model Architecture For each specialized ideal policy, we formulate it using an MLP as the backbone $\mathcal{F}$ of the actor. Perceptive scandots is encoded by an MLP encoder $\mathcal{E}_{\text{terrain}}$. Apart from that, an RNN-based encoder $\mathcal{E}_{\text{hist}}$ is applied to construct the latent information of the history state sequence. Details of model architecture are listed in Table A3.

The actor is formulated as:

$$a_t = \mathcal{F}\left(\mathcal{E}_{\text{hist}}(o_t^{\text{hist}}), \mathcal{E}_{\text{scan}}(o_t^{\text{terrain}}), o_t^{\text{proprio}}, c_t, a_{t-1}\right). \tag{12}$$

The critic is also formulated as a MLP $\mathcal{C}$:

$$v_t = \mathcal{C}\left(\mathcal{E}_{\text{hist}}(o_t^{\text{hist}}), \mathcal{E}_{\text{scan}}(o_t^{\text{terrain}}), o_t^{\text{proprio}}, c_t, a_{t-1}\right). \tag{13}$$

The reinforcement learning algorithm used to train the Goal-conditioned policy is Proximal Policy Optimization Algorithm (Schulman et al., 2017). The hyperparameters of the PPO Algorithm are listed in Table A4 of the supplementary. We use the same hyperparameters for the training of all specialized policies.

Table A3: Specialized Policy Structure

Parameter	Value
Actor Hidden Dims	[512, 256, 128]
Critic Hidden Dims	[512, 256, 128]
Scan Encoder Dims	[128, 64, 32]
RNN Hidden Size	512
RNN Layers	1

Reward Functions for Specialized Policy A walk policy is first trained from scratch through reinforcement learning. Subsequently, a series of terrain-specific policies is developed for individual challenging terrains. To enable this progressive learning capability, we design two distinct reward categories. Basic locomotion rewards, as illustrated in Table A5, which enforces fundamental kinematic constraints and penalizes excessive joint torque/velocity for feasible actuator commands. Apart from that, we design specialized rewards for challenging terrains, as shown in Table A6, to provide harder constraints on various scenarios.

B.3 Details of Autoregressive Diffusion Planner

Terrain-Aware Autoregressive Diffusion Training To bridge the gap between training and autoregressive inference, we employ a three-stage scheduled training (Bengio et al., 2015; Yuan et al.,

Table A4: PPO Hyperparameters for Specialized Policy Training

Hyperparameter	Value
Clip range	0.2
GAE discount factor	0.95
Entropy coefficient	0.01
Max gradient norm	1.0
Learning rate	2e-4
Reward discount factor	0.99
Initial policy std	1.0
Minimum policy std	0.2
Number of environments	2048
Number of environment steps per training batch	24
Learning epochs per training batch	5
Number of mini-batches per training batch	4

Table A5: **Basic Reward Functions and Weight** (Notations: $\boldsymbol{v}$=velocity vector, ω=angular velocity, yaw=heading angle, F=contact force, θ=joint angle, τ=torque, h=height, a=action, $\mathbb{I}(\cdot)$=indicator function)

Reward Function	**Formula**	**Weight**
Orientation	$\sum(g_x^2 + g_y^2)$	-1.0
DOF Acceleration	$\sum \left(\frac{\dot{\theta}_t - \dot{\theta}_{t-1}}{\Delta t}\right)^2$	-2.5e-7
Collision	$\sum \mathbb{I}(\|F_{\text{contact}}\| > 0.1)$	-10.0
Action Rate	$\|a_t - a_{t-1}\|$	-0.1
Delta Torques	$\sum(\tau_t - \tau_{t-1})^2$	-1.0e-7
Torques	$\sum \tau^2$	-0.00001
DOF Error	$\sum(\theta - \theta_{\text{default}})^2$	-0.04
Base Height	$(h_{\text{base}} - h_{\text{target}})^2$	-20.0
Smoothness	$\|a_t - a_{t-1}\| + \|(a_t - a_{t-1}) - (a_{t-1} - a_{t-2})\|$	-0.005
Alive Bonus	$\sum(\text{survival})$	0.005
Tracking Yaw	$\exp(-\|yaw_{\text{target}} - yaw\|^2/0.25)$	0.5
Linear Velocity Z	v_z^2	-1.0
Angular Velocity XY	$\sum(\omega_x^2 + \omega_y^2)$	-0.05

2023; Zhao et al., 2024) curriculum. In our terrain-aware autoregressive diffusion planner, we organize training data into overlapping motion windows of fixed length H and slide them through each sequence. At each training iteration $iter$, we compute a rollout probability p in Alg. 1. We set a maximum of 4 consecutive rollouts during each update to stabilize training and prevent error accumulation.

Algorithm 1 Compute schedule rollout probability

Input: current iteration step $iter$, number of iterations in the fully supervised phase I_1, number of iterations in scheduled phase I_2
Output: probability p

Function SCHEDULE_PROBABILITY($iter$, I_1, I_2)
if $iter \leq I_1$ **then**
    $p \leftarrow 0$ *// Fully supervised phase*
else if $iter > I_1 + I_2$ **then**
    $p \leftarrow 1$ *// Rollout training phase*
else
    $p \leftarrow \frac{iter - I_1}{I_2}$ *// Scheduled training phase*
end if
return p

Table A6: **Special Reward Functions and Weight**.

Reward Function	Formula	Weight
Tracking Goal Velocity	$\min\left(\frac{\boldsymbol{v}_{\text{cur}}\cdot\boldsymbol{v}_{\text{target}}}{\|\boldsymbol{v}_{\text{target}}\|}, v_{\text{cmd}}\right)/v_{\text{cmd}}$	3.0
Hip Position	$\sum(\theta_{hip}-\theta_{\text{default}})^2$	-0.5
Feet Stumble	$\mathbb{I}(\|F_{\text{horizontal}}\| > 4\|F_z\|)$	-1.0
Feet Edge	$\sum \mathbb{I}(\text{foot_on_edge}) \times \mathbb{I}(\text{terrain_level} > 3)$	-1.0
Feet Parallel	$\sum \theta^2_{\text{pitch}}$	-1.0
Huge Step	$\sum\ abs(x_{\text{left_ankle}} - x_{\text{right_ankle}})$	0.001
Feet Yaw	$\sum\ (yaw_{\text{ankle}})^2$	-2.
Feet Dis	$\sum\ ((y_{\text{left_ankle}}) - (y_{\text{right_ankle}}))^2$	-0.001
Knee Dis	$\sum\ ((y_{\text{left_knee}}) - (y_{\text{right_knee}}))^2$	-0.001
Goal	$\sum\ (goal)$	0.0002
Knee Foot	$\sum\ ((y_{\text{knee}}) - (y_{\text{ankle}}))^2$	-0.001
Foot Height	$\sum\ (z_{\text{ankle}})$	0.01
Foot Air Time	$\sum\ (t_{\text{foot_contact==0}})$	10

In our terrain-aware autoregressive diffusion planner, we adopt a complementary three-phase annealing: at each iteration, we sample the history from prior model predictions with probability p and from the ground truth with probability $1-p$, gradually increasing p from 0 to 1:

- **Fully supervised phase.** $p = 0$. Always use ground-truth history, which is fully teacher-forced.
- **Scheduled Training Phase.** p linearly increases from 0 to 1. With probability p, replace the ground-truth history with the model's own rollout history; otherwise keep the ground truth.
- **Rollout Training Phase.** $p = 1$. Always use rollout history, which is fully autoregressive.

This simple curriculum likewise exposes the model to its inference-time distribution early in training, improving long-horizon stability and preventing sequence drift. The training algorithm is shown in Alg. 2.

Implementation Details of Diffusion Models. The denoiser $\mathcal{G}$ is implemented on the DIT backbone with 8 Transformer decoder layers (hidden size 512). We apply classifier-free guidance by masking the terrain embedding with probability 0.1 during both training and inference. Local terrain is encoded from a 187-egocentric height map using lightweight MLP blocks. The final feature map is flattened and projected to a 256-dimensional terrain embedding.

We train for a total of 300K diffusion iterations on our expert locomotion dataset, selecting the best checkpoints at 250K for validation. Optimization is via Adam with batch size 1024. All models run on a single NVIDIA GeForce RTX 4090 GPU for nearly 30 hours. At test time, we use DDPM sampling with guidance scale $\omega = 5$ to autoregressively generate continuous trajectories in real time. Qualitative samples of the generated trajectories are shown in Figure A5.

B.4 Details of HMI-Conditioned Unified Policy

Model Architecture. The network architecture is inherited from Section B.2. Different from a specialized policy network, the goal-conditioned policy network contains an RNN future encoder $\mathcal{E}_{\text{future}}$, which encodes a predicted future state sequence $g_t = [\hat{s}_{t+1}, \ldots, \hat{s}_{t+H}] \in \mathbb{R}^{d\times H}$ sampled from the diffusion planner. The predicted future states are represented in the same body-relative parameterization used by the diffusion and reward function. The architecture of the unified policy is formulated as

$$a_t = \mathcal{F}\left(\mathcal{E}_{\text{hist}}(o_t^{\text{hist}}),\ o_t^{\text{proprio}},\ \mathcal{E}_{\text{future}}(g_t),\ c_t,\ a_{t-1}\right). \tag{14}$$

Reward formulations. Different from the reward functions used during specialized-policy training, the unified policy is trained with an immediate body-space tracking reward together with a long-horizon adversarial reward. Following BeyondMimic (Liao et al., 2025), we adopt a unified task-space

Algorithm 2 Scheduled training for terrain-aware auto-regressive diffusion

Input: denoiser $\mathcal{G}_\theta$ with parameters θ, humanoid state dataloader $\mathcal{X}$, terrain dataloader $\mathcal{Y}$, total diffusion steps N, consecutive window number T, optimizer $\mathcal{O}$, training loss $\mathcal{L}$, max iteration I_{max}, num of iterations in the fully supervised phase I_1, num of iterations in scheduled phase I_2, auto-regressive diffusion sampler $\mathcal{S}$.
$iter \leftarrow 0$
while $iter < I_{max}$ **do**
  $[\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_T] \sim \mathcal{X}, [y_0, y_1, ..., y_T] \sim \mathcal{Y}$ *// Sample T trajectory windows*
  for $t = 1$ to T **do** *// Number of rollouts*
    $\mathbf{x}_t^0 \leftarrow \mathbf{x}_t$
    $n \sim \mathcal{U}[0, N)$
    $\mathbf{x}_t^n \leftarrow$ FORWARD_DIFFUSION$(\mathbf{x}_t^0, n)$
    $\hat{\mathbf{x}}_t^0 = \mathcal{G}_\theta(\mathbf{x}_t^n, \mathbf{x}_{t-1}, t, y_t)$ *// Next state denoising*
    $\nabla \leftarrow \nabla_\theta \mathcal{L}(\mathbf{x}_t^0, \hat{\mathbf{x}}_t^0)$
    $\theta \leftarrow \mathcal{O}(\theta, \nabla)$ *// Back propagation*
    $p \leftarrow$ SCHEDULE_PROBABILITY$(iter, I_1, I_2)$ *// Sample schedule prob*
    if rand$() < p$ **then**
      $\mathbf{x}_t^N \leftarrow$ FORWARD_DIFFUSION$(\mathbf{x}_t^0, N)$ *// Maximum noising*
      $\tilde{\mathbf{x}}_t^0 = \mathcal{S}(\mathcal{G}_\theta, \mathbf{x}_t^N, \mathbf{x}_{t-1}, N, y_t)$ *// Full trajectory sampling*
      $\mathbf{x}_t \leftarrow \tilde{\mathbf{x}}_t^0$ *// Set predicted state into history*
    else
      $\mathbf{x}_t \leftarrow \mathbf{x}_t^0$ *// Use dataset state history*
    end if
    $iter \leftarrow iter + 1$
  end for
end while

body-tracking formulation for the immediate reward. Specifically, let $\mathcal{B}_{\text{target}}$ denote the set of tracked bodies, and let the desired body poses and twists be extracted from the first predicted step $\hat{s}_{t+1}$ of the diffusion planner. In practice, these desired quantities can be read directly from $\hat{s}_{t+1}$ or computed from it via forward kinematics in the body-relative frame.

For each target body $b \in \mathcal{B}_{\text{target}}$, we define

$$\mathbf{e}_{p,b} = \mathbf{p}_b^{\text{des}} - \mathbf{p}_b, \qquad \mathbf{e}_{R,b} = \log\left(R_b^{\text{des}} R_b^\top\right), \tag{15}$$

$$\mathbf{e}_{v,b} = \mathbf{v}_b^{\text{des}} - \mathbf{v}_b, \qquad \mathbf{e}_{\omega,b} = \boldsymbol{\omega}_b^{\text{des}} - \boldsymbol{\omega}_b. \tag{16}$$

The mean squared error of each tracking term is averaged over all target bodies:

$$\bar{e}_s = \frac{1}{|\mathcal{B}_{\text{target}}|} \sum_{b \in \mathcal{B}_{\text{target}}} \|\mathbf{e}_{s,b}\|^2, \qquad s \in \{\mathbf{p}, R, \mathbf{v}, \boldsymbol{\omega}\}. \tag{17}$$

Each term is then mapped through a Gaussian-shaped score:

$$r(\bar{e}_s, \sigma_s) = \exp\left(-\bar{e}_s / \sigma_s^2\right), \tag{18}$$

and the immediate tracking reward is defined as

$$r_t^{\text{track}} = \sum_{s \in \{\mathbf{p}, R, \mathbf{v}, \boldsymbol{\omega}\}} r(\bar{e}_s, \sigma_s). \tag{19}$$

To complement the body-tracking objective, we introduce a modified AMP-style (Peng et al., 2021) adversarial reward. Different from standard AMP, the discriminator is updated using future motion plans online-sampled from the diffusion planner rather than pre-recorded motion clips, as described in Algorithm 3. Therefore, r_t^{track} provides immediate body-level geometric supervision, while r_t^{style} encourages temporally consistent and terrain-adaptive whole-body motion.

Reward Functions and Weight. Table A8 lists the effective scalar reward weights used for unified policy learning. The term "effective weight" denotes the coefficient directly applied to each scalar

Table A7: Immediate Body-space Tracking Reward for Unified Policy.

Tracking Term	Score	Tolerance
Position	$\exp(-\bar{e}_p/\sigma_p^2)$	σ_p
Orientation	$\exp(-\bar{e}_R/\sigma_R^2)$	σ_R
Linear Velocity	$\exp(-\bar{e}_v/\sigma_v^2)$	σ_v
Angular Velocity	$\exp(-\bar{e}_\omega/\sigma_\omega^2)$	σ_ω
Total	$\sum_{s\in\{\mathbf{p},R,\mathbf{v},\boldsymbol{\omega}\}} \exp(-\bar{e}_s/\sigma_s^2)$	–

Algorithm 3 Training with Online Diffusion AMP

Input $\mathcal{M}_{\text{diff}}$: autoregressive diffusion planner, K: history length, H: horizon
initialize discriminator D, policy π, value function V, replay buffer $\mathcal{B} \leftarrow \emptyset$, history buffer $\mathcal{H} \leftarrow \emptyset$.
while not done **do**
  for trajectory $i = 1, ..., m$ **do** *// Data collection phase*
    $\tau^i \leftarrow \emptyset$
    for step $t = 0, \ldots, T-1$ **do**
      Execute $a_t \sim \pi(s_t)$, observe s_{t+1}, store $(s_t, a_t, r_t^{\text{regu}}, s_{t+1})$ in $\mathcal{B}$
      Update $\mathcal{H}$ with s_t: $\mathcal{H} \leftarrow \text{concat}(\mathcal{H}[-K+1:], s_t)$ *// Maintain latest K states*
      if $t < K-1$ **then** *// Padding for initial phase*
        $\hat{\mathcal{H}} \leftarrow [s_0,\ ...,\ s_0, \mathcal{H}[0:t+1]]$ *// Pad with s_0*
      else
        $\hat{\mathcal{H}} \leftarrow \mathcal{H}[-K:]$ *// Last K states*
      end if
      Generate $\{\hat{s}_{t+1:t+H}\} \sim \mathcal{M}_{\text{diff}}(\hat{\mathcal{H}})$
      $d_t \leftarrow D(\Phi(s_t), \Phi(\hat{s}_{t+1}))$ *// Use immediate prediction*
      $r_t^{\text{style}} \leftarrow$ style reward via Eq.10 using d_t
      $r_t \leftarrow w_{\text{regu}} r_t^{\text{regu}} + w_{\text{track}} r_t^{\text{track}} + w_{\text{style}} r_t^{\text{style}}$
      Append (s_t, a_t, r_t, s_{t+1}) to τ^i
    end for
  end for
  for update step $= 1, ..., n$ **do** *// Training phase*
    $b^\pi \leftarrow$ sample K transitions $\{(s_j, s'_j)\}_{j=1}^K$ from $\mathcal{B}$
    Generate $\{\hat{s}_{j:j+H}\}_{j=1}^K \sim \mathcal{M}_{\text{diff}}(b^\pi)$ *// On-the-fly generation*
    Update D via Eq.9 using $\{(\Phi(s_j), \Phi(\hat{s}_{j+1}))\}_{j=1}^K$ *// Fresh samples*
    Update V and π using $\{\tau^i\}_{i=1}^m$ *// Policy optimization*
  end for
end while

reward before summation in Eq. 7. The reward design contains 19 scalar terms in total: 4 body-space tracking terms, 10 physical regularization terms, 4 safety terms, and 1 diffusion-AMP style term. Compared with the specialized policies, we remove terrain-specific shaping rewards and keep only embodiment-level regularization, because terrain-dependent motion targets are provided by the diffusion future trajectory $g_t = \hat{s}_{t+1:t+H}$.

For the tracking terms, the body-space errors follow the definitions in Eq. 19. Specifically, the desired body pose and twist are extracted from the first predicted state $\hat{s}_{t+1}$ generated by the diffusion planner. The four tracking terms provide immediate geometric supervision, while the diffusion-AMP term in Eq. 10 regularizes the policy toward temporally consistent motion patterns over the generated future trajectory. Therefore, the unified reward remains compact while still capturing terrain-adaptive whole-body locomotion behavior.

DreamPolicy Inference Process. The inference process is addressed in Algorithm 4: the agent first observes the current state s_t from the environment and acquires the terrain heightmap, stores the

Table A8: **Reward Functions and Weights for the Unified Policy.** Notation: $\mathcal{B}_{\text{target}}$ denotes the tracked body set; q, $\dot{q}$, $\ddot{q}$ denote joint position, velocity, and acceleration; τ denotes joint torque; a_t denotes policy action; c_f is the contact indicator of foot f; $\mathbf{g}^{\text{proj}}$ is the projected gravity vector; $\mathbb{I}(\cdot)$ is the indicator function.

Reward Function	**Formula**	**Weight**
Diffusion-guided Tracking Reward		
Body Position Tracking	$\exp\left(-\bar{e}_{\mathbf{p}}/\sigma_{\mathbf{p}}^2\right)$	1.0
Body Orientation Tracking	$\exp\left(-\bar{e}_{R}/\sigma_{R}^2\right)$	1.0
Body Linear Velocity Tracking	$\exp\left(-\bar{e}_{\mathbf{v}}/\sigma_{\mathbf{v}}^2\right)$	1.0
Body Angular Velocity Tracking	$\exp\left(-\bar{e}_{\boldsymbol{\omega}}/\sigma_{\boldsymbol{\omega}}^2\right)$	1.0
Physical Regularization Reward		
Linear Velocity Z	v_z^2	-1.0
Base Angular Velocity XY	$\|\boldsymbol{\omega}_{xy}\|_2^2$	-0.05
Hip Joint Deviation	$\sum_{j\in\mathcal{J}_{\text{hip}}}(q_j - q_{j,\text{default}})^2$	-0.5
Joint Torques (L_2)	$\|\boldsymbol{\tau}\|_2^2$	$-1.5\text{e}{-7}$
Joint Acceleration (L_2)	$\|\ddot{\mathbf{q}}\|_2^2$	$-1.25\text{e}{-7}$
Joint Velocity (L_2)	$\|\dot{\mathbf{q}}\|_2^2$	$-1.0\text{e}{-4}$
Action Rate (L_2)	$\|a_t - a_{t-1}\|_2^2$	-0.005
Flat Orientation	$\|\mathbf{g}_{xy}^{\text{proj}}\|_2^2$	-3.0
Feet Slide	$\sum_f \|\mathbf{v}_{xy,f}\|_2 \cdot \mathbb{I}(c_f)$	-0.4
Energy Consumption	$\sum_j(\tau_j\dot{q}_j/k_j)^2$	$-5.0\text{e}{-5}$
Safety Reward		
Joint Position Limits	$\sum_j\left(\max(0, q_j - q_{j,\max}) + \max(0, q_{j,\min} - q_j)\right)$	-1.0
Joint Velocity Limits	$\sum_j \max(0, \lvert\dot{q}_j\rvert - 0.9\dot{q}_{j,\max})$	-1.0
Torque Limits	$\sum_j \max(0, \lvert\tau_j\rvert - 0.8\tau_{j,\max})^2$	-0.01
Undesired Contacts	$\mathbb{I}\left(\text{count}(\text{collision}_{\text{body}\setminus\text{feet}}) > 0\right)$	-1.0
Diffusion-AMP Style Reward		
AMP Style	$\max\left\{0,\ 1 - 0.25\left(D(\Phi(s_t), \Phi(s_{t+1})) - 1\right)^2\right\}$	0.25

current state in the history buffer, then interacts with the environment to obtain the action, executes the action, and finally transitions to the next state $s_{\text{t+1}}$.

C Additional Experiments and Results

C.1 Detailed Implementation of Baselines

Scenario Specialized Policy. Inspired by (Cheng et al., 2024c), we use a scandot encoder and a privileged information estimator, as mentioned in Section B.2. To this end, we will not use this setting for experiments on multi-terrains.

Distillation-based Method. Inspired by (Zhuang et al., 2023; Zhang et al., 2025a; He et al., 2024a), which trained a unified policy using online distillation via DAgger (Ross et al., 2011). We trained a policy via online distillation, where the unified policy learn from multiple single specialized policies via supervision.

Joint Multi-terrain Policy. This baseline is trained with the same configuration as the scenario specialized policy, i.e., the same network architecture, observation space, reward design, curriculum, and training budget, except that a single policy is trained jointly on our final mixed terrain that combines all the primitive terrains, instead of on a single terrain.

Ablation on Diffusion Planner. In this setting, we remove the autoregressive diffusion planner. Due to the lack of motion guidance, we train the discriminator using the standard implementation of AMP (Peng et al., 2021), in which we sample transitions from offline trajectories collected from specialized policies.

Algorithm 4 DreamPolicy Inference

Input $\mathcal{M}_{\text{diff}}$: trained diffusion planner, π: trained policy, K: history length, H: horizon
$\mathcal{H} \leftarrow \emptyset$ initialize history buffer
$z_{\text{terrain}} \leftarrow$ preprocessed terrain heightmap
while running **do**
  Observe current state s_t
  Update $\mathcal{H}$: $\mathcal{H} \leftarrow \text{concat}(\mathcal{H}[-K+1:], s_t)$ *// Maintain latest K states*
  if $|\mathcal{H}| < K$ **then** *// Initial phase padding*
    $\hat{\mathcal{H}} \leftarrow [s_0, \ ..., \ s_0, \mathcal{H}[0:t+1]]$ *// Pad with initial state*
  else
    $\hat{\mathcal{H}} \leftarrow \mathcal{H}[-K:]$ *// Last K states*
  end if
  Generate $\{\hat{s}_{t+1:t+H}\} \sim \mathcal{M}_{\text{diff}}(\hat{\mathcal{H}}, z_{\text{terrain}})$ *// Terrain-conditioned prediction*
  Construct policy input: $\tilde{s}_t \leftarrow [s_t; \hat{s}_{t+1:t+H}]$ *// Concatenate future predictions*
  $a_t \leftarrow \pi(\tilde{s}_t)$ *// Generate action*
  Execute a_t on robot and observe next state s_{t+1}
end while

Ablation on Goal-Conditioned RL. Here we remove the goal-conditioned tracker and let the diffusion model control the robot end-to-end, i.e., the direct-action diffusion policy setting. A predicted state sequence is not executable without a tracker, so controlling the robot requires emitting joint targets; we therefore retrain the model with actions as the diffusion target. Action sequences are shaped by terrain, contact and command variation and vary far more sharply than the dynamics-governed states, making them a harder generative target (Lu et al., 2025; Ajay et al., 2023). This ablation performs poorly on complex terrains (C.2), consistent with the benefit of conditioning the policy on diffusion-generated state references.

C.2 Additional Experiments

Table A9 shows the detailed data on a single terrain that we display using lidar maps in Figure A3. The detailed data shows that our method does not exhibit significant performance degradation.

Table A9: **Settings Comparison in Single-Terrain**. All data in the table are shown as percentages; the higher the better. The best results are highlighted in boldface.

	Uneven		Bridge		Stair		Flat		Gap	
	R_{succ} (%)	R_{cmp} (%)	R_{succ} (%)	R_{cmp} (%)	R_{succ} (%)	R_{cmp} (%)	R_{succ} (%)	R_{cmp} (%)	R_{succ} (%)	R_{cmp} (%)
Specialized Policy	98.34	99.21	98.82	99.33	**99.42**	**99.82**	99.50	99.80	99.27	**99.78**
Distillation	95.33	98.02	98.31	99.29	94.65	95.21	99.41	99.78	98.78	99.31
Joint multi-terrain Policy	88.62	92.94	91.14	95.18	89.26	93.58	93.18	96.71	86.83	91.47
Ours	**99.37**	**99.70**	**99.65**	**99.79**	98.46	99.33	**99.58**	**99.85**	**99.39**	99.61
Ours w/o Diffusion Planner	30.25	41.45	90.11	92.36	87.64	91.61	**99.64**	99.74	88.53	92.57
Ours w/o GCRL	0.06	1.41	60.32	64.42	35.08	39.18	96.34	98.12	50.86	62.35

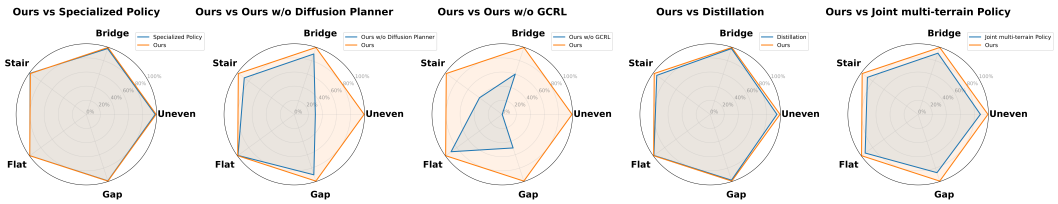


Figure A3: **Comparison of success rates across single terrains.** This figure compares the DreamPolicy against five baselines across five single terrain types. Each subplot contrasts Ours with one baseline method, with radial axes representing success rates.

C.3 Primitive Terrain Analysis

The t-SNE visualization (Fig. A4) reveals a structured latent space where the **Plane** (Blue) acts as a central hub, with complex terrains forming distinct, specialized clusters. Kinematically, these clusters represent conflicting motion modes: **Stairs** (Purple) demand high hip flexion for clearance, while **Stepping Stones** (Yellow) require precise, discrete foot placement.

Conventional distillation fails here by collapsing these multi-modal distributions into a mean representation, causing instability in mixed scenarios. In contrast, our framework treats these clusters as composable primitives. By learning to smoothly interpolate between these distinct kinematic modes rather than averaging them, our method enables robust, zero-shot adaptation to unseen composite terrains.

More simulation results are shown in Figures A2 and A1

Additional real-world visualization is shown in Figure A6.

C.4 Real-world Deployment Details

The deployment follows the distributed ROS architecture described in the main text, which keeps the observation pipeline on hardware aligned with the one used in training. Raw LiDAR from the onboard computer is processed by elevation mapping, downsampled and cropped to a robot-centric height grid ($[-0.6, 1.0]$ m longitudinal, ± 0.5 m lateral), and published at $\approx$20 Hz; inference on the external PC adds $\approx$20 ms of latency, which the asynchronous buffer hides from the 50 Hz control loop.

Adaptability to composite terrains. A key challenge in our experiments is the transition between different terrain types (e.g., from a flat platform to a stair, or across mixed obstacles). Here the diffusion planner plays a critical role. By synthesizing terrain-adaptive reference trajectories, the system anticipates future kinematic states, allowing the policy to adjust the robot's torso posture and foot placement. This predictive capability is crucial for maintaining balance on composite terrains.

Dominant failure modes. Across the quantitative real-world trials reported in the main text, the dominant failure cases arise from perception noise and inaccurate terrain reconstruction, particularly around terrain edges, together with sim-to-real discrepancies in contact dynamics, including differences in friction and foot slippage.

D Supplementary Video

We have included a video in the supplementary material that summarizes our main contributions and presents video results of humanoid locomotion. More details are as follows:

- We introduced our DreamPolicy algorithmic framework, summarized our main contributions across four aspects, and detailed each component.
- We presented our quantitative results on single-terrain and unseen terrains and analyzed the impact of scale on our method.
- We gave a visual comparison of our performance on different terrains with the traditional distillation method, showing that our algorithm has significant advantages.

E Extended Limitation and Discussions

E.1 Limitation and Failure Cases

Our analysis suggests that the limitations may stem from two factors: 1) **Naive terrain encoding.** The current diffusion planner employs a plain MLP to process the 187-dimensional structured terrain elevation map, without any spatially aware components that could extract local geometric patterns and contextual cues essential for handling sharp gradients or complex surface discontinuities. 2) **Limited policy perception.** Our unified policy relies exclusively on the planner's state trajectory

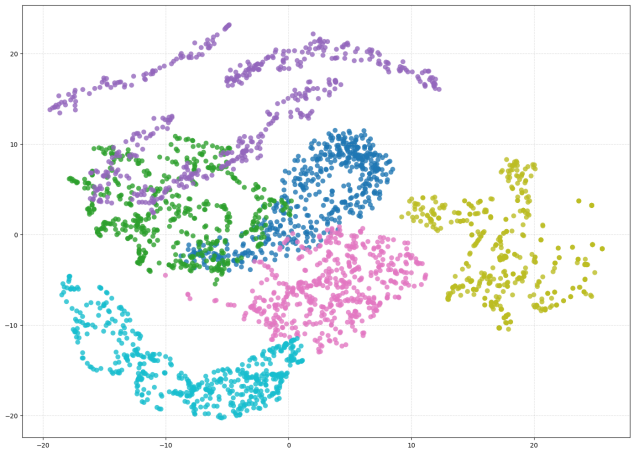


Figure A4: Terrain t-SNE visualization.

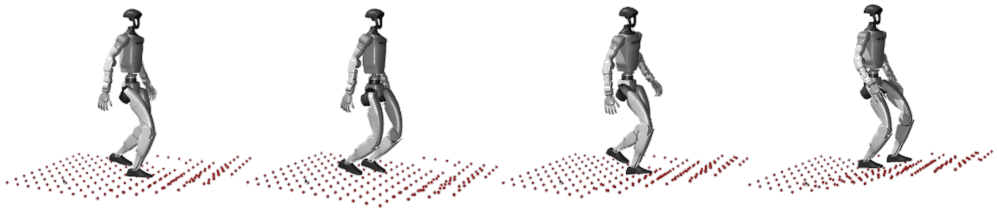


Figure A5: **Visualization of Terrain-Adaptive Motion.** The generated trajectories demonstrate the model's ability to proactively anticipate terrain changes, such as adjusting step height in advance for stairs.

predictions for navigation, lacking any direct perception of the terrain. Consequently, the policy is unable to react to unforeseen geometric perturbations or dynamically adjust its foot placement in extreme, unstructured environments.

These two limitations, inadequate feature representation of challenging terrain and lack of real-time sensory feedback, undermine stability on terrain outside the training distribution.

E.2 Why Composition Works, and Where It Stops

Our composition ability follows from the support of the expert data rather than from any explicit composition operator. Terrains are not kinematically disjoint: a stair expert and a slope expert both contain upright stance, periodic foot placement, and contact transitions, so expert rollouts overlap in state space and their union forms one connected, multi-modal distribution, which the planner interpolates within rather than selects from; an interpolation between two feasible motions is itself approximately feasible, which is why an unseen stair–bridge terrain is tractable. The push perturbations applied during training widen the same modes with small recovery excursions, so recovery references are available as well. The mechanism stops at the support of that union: a terrain demanding a motion mode no expert has exhibited forces the planner to extrapolate, and the extrapolated reference need not be dynamically feasible. Our unified policy has no terrain perception of its own and cannot detect this; it keeps tracking a reference the robot cannot realize, which is the failure case discussed in Section E.1.

E.3 Discussion and Future Work

We propose a set of comprehensive research directions and rigorous validation studies that span dataset augmentation, terrain-generalization techniques, inference optimization, and real-world robotic trials, in order to systematically extend the capabilities and verify the practical effectiveness of our diffusion-based humanoid motion planning framework:



(a) Outdoor qualitative results for upstairs scenario.



(b) Outdoor qualitative results on upstairs scenario.



(c) Outdoor qualitative results on downstairs scenario.



(d) Outdoor qualitative results on long upstairs slope.



(e) Outdoor qualitative results on grass scenario.



(f) Indoor qualitative results on an upward slope.



(g) Indoor qualitative results for downstairs scenario.



(h) Indoor qualitative results for upstairs scenario.

Figure A6: Qualitative results from real-world demonstrations in outdoor and indoor environments.

- **Augment dataset diversity.** Integrate additional data sources beyond the current set of expert policies trained in RL, including demonstrations from varied control algorithms, motion capture recordings of human subjects, and synthetic trajectories generated under different environmental perturbations, thereby significantly enriching the breadth, variability, and robustness of the humanoid motion dataset.
- **Enhance terrain generalization.** Develop and evaluate advanced terrain-encoding strategies, such as learned geometric embeddings, height-map spectral features, and multimodal descriptors, that can effectively generalize across a wide spectrum of previously unseen environments, thereby enabling the diffusion planner to produce reliable, accurate future-state priors even on novel or complex terrains.
- **Accelerate inference efficiency.** Reduce sampling latency by integrating techniques such as step-reduction schedulers, diffusion model distillation into lightweight student networks, dynamic stopping criteria based on convergence metrics, or adaptive iteration strategies that allocate computation where it is most needed, all aimed at supporting true real-time deployment on computationally constrained hardware.
- **Standardize benchmarking.** Develop comprehensive, open evaluation suites comprising diverse task scenarios, standardized test terrains, and reproducible performance metrics (e.g. success rate, energy consumption, and robustness measures), in order to facilitate fair, transparent comparisons across diffusion-based humanoid planners and to drive accelerated community progress.