Perceptive Behavior Foundation Model: Adapting Human Motion Priors to Robot-Centric Terrain

Anonymous authors

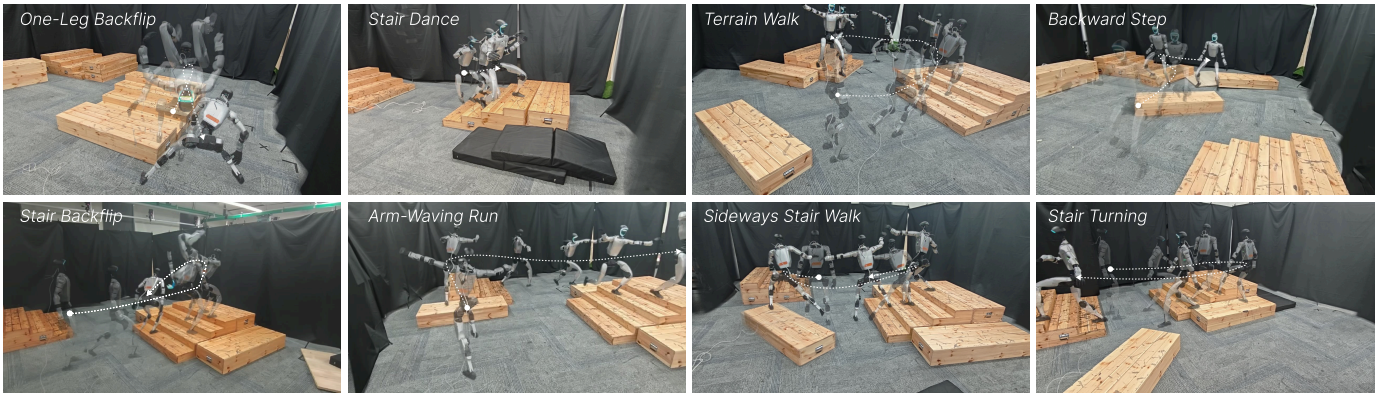

Fig. 1. **Single-policy terrain grounding.** A single Perceptive BFM tracks diverse flat-ground human-motion commands while adapting them to randomly placed robot-side terrains. Robot-centric perception adjusts footholds, swing clearance, posture, and contact timing online.

***Abstract*—Humanoid behavior foundation models aim to acquire reusable whole-body control policies from broad human motion priors, enabling a single controller to produce diverse and expressive behaviors. However, existing motion-centric foundation policies largely assume that the reference motion is already physically compatible with the robot's surroundings. This assumption breaks when the demonstrator, operator, and robot inhabit different environments: a human motion may specify the intended behavior, but not the footholds, clearance, body height, or contact timing required by the robot's local terrain. We introduce *Perceptive Behavior Foundation Model* (Perceptive BFM), a terrain-aware humanoid control framework that grounds human motion priors in robot-centric perception. The model preserves raw kinematic motion references as the behavioral interface, while using local terrain observations to adapt contacts, posture, and timing. To provide scalable terrain supervision, we develop *terrain-conformal reference synthesis* (TCRS), which converts locomotion-oriented human motion clips into terrain-consistent references through contact-aware foothold construction, foot-geometry-aware swing optimization, support-aware root reconstruction, collision repair, and multi-point inverse kinematics. We then train a blind adapted-reference teacher and transfer its terrain-conformal behavior to a deployed raw-reference student through target-frame action alignment. The student is an identity-gated Transformer tracker whose terrain features enter through residual pathways initialized to preserve the motion-tracking prior and trained to produce local corrections only when needed. Across quantitative TCRS analysis and end-of-training logged rollout diagnostics, the TCRS-supervised run reports higher logged closed-loop tracking success ($27.3\% \rightarrow 55.1\%$) under the same final-PPO setup. In simulation, cross-terrain rollouts evaluate stairs, slopes, sparse supports, recessed obstacles, and mixed indoor terrain; on hardware, the same policy qualitatively demonstrates locomotion, stylistic motions, acrobatic maneuvers, and motion-capture teleoperation across indoor, outdoor, and mocap-mismatch settings. The results suggest that robot-centric perception can turn human motion priors into terrain-compatible whole-body behavior without changing the raw motion command interface.**

I. Introduction

Humanoid control is rapidly shifting from isolated, task-specific skills toward generalist behavior foundation models and large motion-tracking policies. Recent whole-body trackers reproduce diverse motions with a single learned controller [11, 2, 9], while newer foundation-control systems further scale behavior priors, command encoders, policy capacity, and downstream interfaces [22, 36, 15, 19]. This trend is important because it turns human motion into a reusable command interface: instead of designing a separate controller for each maneuver, the robot can learn a broad prior over human-like whole-body intent.

We use the term behavior foundation model in the locomotion-oriented sense: a reusable motion-reference interface and broad behavior prior over humanoid motions, not a claim of open-ended coverage over all humanoid skills.

This progress, however, exposes a hidden assumption: most reference-centric formulations ask how accurately the robot can reproduce the supplied motion, not how that motion should be physically grounded in the robot's own environment. These are different problems. A flat-ground walking reference does not specify stair footholds; a teleoperator in a control room does not encode sparse supports or recessed terrain

at a remote site; a clean-floor demonstration does not tell the robot how much swing clearance is needed to avoid obstacles. The reference conveys intent and style, but it may not be a terrain-valid trajectory in the robot's local world. This is the operator–environment mismatch: the human supplies the desired behavior, while the robot must resolve terrain-specific contacts, body height, balance, and timing from its own perception.

Existing work leaves a gap between two successful directions. General motion-tracking and behavior-foundation models excel at diverse, expressive whole-body behavior, but their command interfaces obscure the need for environment-conditioned contact adaptation. Perceptive locomotion and parkour policies use height maps, depth, and terrain observations to traverse obstacles or sparse footholds [39, 42, 34], but they are organized around traversal skills, motion matching, or system-selected maneuvers rather than preserving an arbitrary human motion command. The missing capability is the perceptual grounding of behavior priors: using robot-centric terrain perception to reinterpret what a human motion command physically requires.

We introduce *Perceptive Behavior Foundation Model* (Perceptive BFM), a single-policy framework for terrain-aware humanoid motion tracking. In this work, the foundation interface is the kinematic motion reference: a unified command representation that lets one policy reuse broad whole-body motion priors while grounding their terrain-dependent realization in robot-centric perception. The underlying control problem is *perceptive motion tracking*: given a raw kinematic reference, robot proprioception, and local terrain observation, the policy must generate whole-body actions that are both behaviorally faithful and environmentally feasible. The raw reference remains the deployment command. Terrain perception provides only the local realization: footholds, clearance, posture, and contact timing.

The method is built around a staged *Perceptive Motion Tracking* (PMT) training algorithm (Figure 2). First, an offline *terrain-conformal reference synthesis* (TCRS) module converts raw motion clips and sampled height fields into terrain-consistent supervision. Rather than presenting this module as a pair of low-level optimizers, we formulate it as structured reference synthesis: contact-aware foothold construction, foot-geometry-aware swing optimization in a mid-foot frame, support-aware root reconstruction, collision repair, and multi-point Jacobian IK. Second, a blind Transformer teacher learns to track the synthesized terrain-conformal references. Third, a vision-conditioned student receives the original raw reference and a local terrain scan, and imitates the teacher through target-frame action alignment, which expresses the teacher's effective joint target in the student's raw-command frame. Finally, PPO fine-tunes the student with identity-gated terrain residuals, updating the transferred tracking prior conservatively while learning local perception-conditioned corrections.

The architecture follows the same separation of roles. Command and proprioceptive histories form a motion-tracking latent, while terrain observations enter through zero-initialized intent and action-residual pathways, so at initialization the student behaves as a raw-reference tracker and terrain features only contribute through residual branches learned during distillation and fine-tuning. Our experiments combine quantitative and deployment evidence: reference-quality metrics isolate TCRS supervision, end-of-training logged rollout diagnostics compare component choices under the same final-PPO setup, cross-terrain simulation rollouts evaluate fixed policies across terrain families, and real-robot indoor, outdoor, and motion-capture mismatch rollouts provide qualitative deployment coverage. Figure 1 shows the same policy accepting diverse flat-ground commands and adapting each to a randomly placed terrain layout.

Contributions. This paper introduces *Perceptive BFM*, a motion-reference-conditioned humanoid behavior foundation model that grounds human motion priors in robot-centric terrain. We make three contributions:

- We introduce *TCRS*, a scalable offline synthesis pipeline that converts raw human motion and sampled height fields into terrain-consistent supervision through contact-aware foothold construction, foot-geometry-aware swing optimization, support-aware root reconstruction, collision repair, and multi-point leg IK.
- We propose *PMT*, a raw-reference teacher–student algorithm for deploying terrain-aware behavior without changing the command interface. A blind teacher tracks TCRS references, while a vision student receives the original raw reference; target-frame action alignment transfers the teacher's terrain-conformal behavior into the student's raw-reference action frame.
- We design and evaluate an identity-gated Transformer policy for single-policy terrain grounding of broad human motion priors. The policy initializes as a raw-reference tracker and learns terrain-conditioned residual corrections from robot-centric perception, enabling the same command interface to support locomotion, expressive motions, acrobatics, and mocap-based operator–environment mismatch across diverse robot terrains.

II. Related Work

a) Generalist humanoid behavior and expressive tracking.: DeepMimic and AMP established reinforcement-learning approaches for tracking motion clips and imitation priors [24, 25]. Recent humanoid systems scale these ideas to richer corpora and robot embodiments: H2O and OmniH2O learn teleoperation and universal tracking policies [8, 11]; expressive whole-body controllers reproduce diverse human motions on hardware [2]; and PHC, PULSE, HOVER, HumanPlus, reference-guided motion tracking, OmniXtreme, and SONIC broaden the representation, command encoder, model scale, or residual refinement stack [17, 18, 9, 7, 20, 30, 19]. Robot foundation and behavior-foundation models push the same trend toward reusable whole-body policies and promptable control [22, 36, 15]. Perceptive BFM builds on this tracking tradition but targets a complementary failure mode: broad motion priors can encode human intent without specifying terrain-valid contacts in the robot's world.

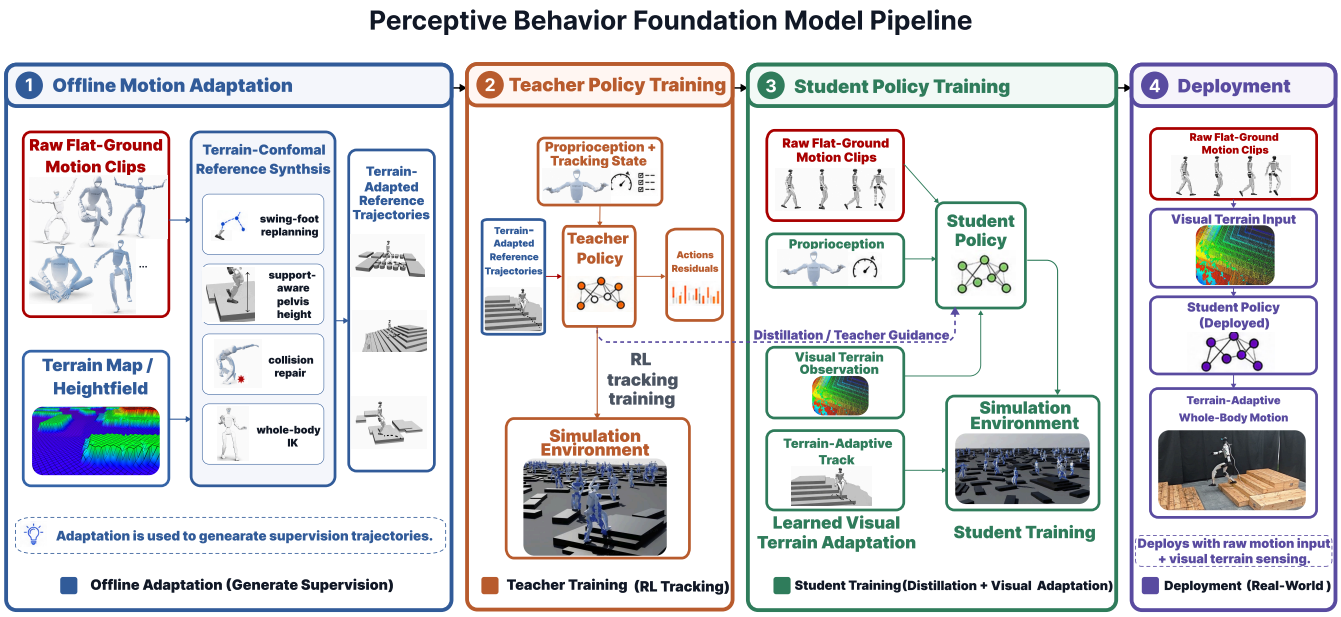

Fig. 2. **Perceptive BFM overview.** TCRS synthesizes terrain-conformal references *offline only*; it is never queried at deployment. A blind teacher learns adapted-reference tracking on this supervision; the deployed identity-gated Transformer student receives the raw reference and a robot-centric terrain scan, and learns local residual corrections through target-frame action alignment. The deployment command remains the raw kinematic reference.

b) Perceptive locomotion and terrain-aware whole-body control.: Terrain-aware legged policies use height maps, depth, or images to traverse obstacles and sparse footholds, avoid collisions, and maintain safe, comfortable locomotion in dynamic environments [1, 40, 3, 41, 29, 27, 39, 32, 33]. Recent whole-body parkour systems integrate exteroception into motion tracking or distill depth policies from expert controllers, enabling contact-rich skills such as climbing, vaulting, rolling, or obstacle traversal [42, 34]. These systems demonstrate the value of perception for terrain interaction. Our setting is stricter in a different sense: the user-provided motion remains the behavior to preserve, so perception should ground the command rather than replace it with a generic terrain skill or an autonomously selected parkour maneuver.

c) Generated, repaired, and terrain-conditioned references.: A closely related interface lets the system choose, generate, or repair a feasible reference. Generator–tracker systems synthesize terrain-conditioned motions online [37]; navigation-oriented reference-guided RL modulates trajectories to be consistent with terrain geometry [5]; motion-matching parkour retrieves and chains skill clips before distilling a perceptive controller [34]; and physics-consistent tracking pipelines roll out privileged policies to filter infeasible references before training a deployable tracker [16]. These methods are strong when feasibility, reference repair, or autonomous skill selection is the primary objective. We instead keep the raw clip fixed at deployment and use terrain-conformal references only as offline supervision for a raw-reference student.

d) Retargeting, reference synthesis, and residual distillation.: Human-to-robot retargeting and trajectory optimization convert demonstrations into robot-executable references [4, 28, 35, 6, 23]. Teacher–student training, curriculum learning, and residual policies are common tools for transferring privileged or easier-to-train behavior into deployable policies [14, 13, 21, 10, 26, 12, 38, 31]. Our use of these tools is interface-specific: TCRS synthesizes terrain-conformal supervision, the teacher tracks that adapted reference, the student receives the original raw reference, and identity-gated residuals make terrain correction an explicitly learned departure from the raw-command tracker.

III. Method

Perceptive BFM is trained with a staged *PMT* algorithm. The key interface contract is that the raw motion reference remains the deployment command: terrain-conformal references are used to supervise learning, but they are not supplied to the final policy at test time. PMT has four stages. Stage 1 synthesizes terrain-conformal references offline with TCRS. Stage 2 trains a blind teacher on those synthesized references. Stage 3 distills a vision student that receives the raw reference and robot-centric terrain observation. Stage 4 fine-tunes the student with PPO while updating the transferred motion-tracking prior conservatively.

A. Problem Formulation

Let $\mathbf{o}_t^{\text{prop}}$ denote robot proprioception, $\mathbf{o}_t^{\text{vis}}$ denote the local terrain observation, and $\mathbf{m}_{t:t+H}^{\text{raw}}$ denote a future window of the raw kinematic reference. In our implementation, the deployment command is represented by target joint positions and velocities, together with local motion-anchor displacements used by the tracker. Perceptive motion tracking asks for a policy

$$\mathbf{a}_t = \pi_\theta\left(\mathbf{o}_t^{\text{prop}}, \mathbf{o}_t^{\text{vis}}, \mathbf{m}_{t:t+H}^{\text{raw}}\right), \tag{1}$$

whose action preserves the behavior encoded in the reference while adapting contact, posture, and local feasibility to the robot's terrain. The policy outputs a residual joint-position target

$$\mathbf{q}_t^{\mathrm{pd}} = \mathbf{q}_t^{\mathrm{cmd}} + \mathbf{a}_t, \tag{2}$$

where $\mathbf{q}_t^{\mathrm{cmd}}$ is the command-frame joint reference. The teacher uses $\mathbf{q}_t^{\mathrm{cmd}} = \mathbf{q}_t^{\mathrm{tcrs}}$, the terrain-conformal reference synthesized offline. The deployed student uses $\mathbf{q}_t^{\mathrm{cmd}} = \mathbf{q}_t^{\mathrm{raw}}$. This convention makes the transfer problem explicit: the teacher learns to track an adapted command, while the student must express the same terrain correction around the unmodified user command.

The deployable actor receives projected gravity, base angular velocity, joint positions, joint velocities, previous actions, a 10-step proprioceptive history, a 21-step reference-command window, and a 21-step motion-anchor displacement window. Terrain perception is a torso-centered ray-cast height scanner over a 1.6 m $\times$ 1.0 m region at 0.1 m resolution, represented as a normalized 17×11 height map with a validity mask. Privileged ground-truth body-pose, global-anchor, and base-linear-velocity terms are used only by the teacher, critic, or auxiliary losses, and are removed from the deployable student actor, which instead reads detached estimator outputs (Appendix A). The full PMT training pipeline is given in Appendix A (Alg. 1): TCRS produces paired $(\mathbf{m}^{\mathrm{raw}}, \mathbf{m}^{\mathrm{tcrs}}, \tau)$ data, a blind teacher trains with PPO on $\mathbf{m}^{\mathrm{tcrs}}$, an identity-gated vision student is distilled with target-frame labels from the teacher, and the student is fine-tuned with PPO under raw-reference commands.

B. Terrain-Conformal Reference Synthesis

A raw human motion is behaviorally informative but not necessarily terrain-conformal when placed in the robot's environment. We therefore introduce a synthesis operator

$$\mathbf{m}_{1:T}^{\mathrm{tcrs}} = \mathcal{S}_{\mathrm{TCRS}}\left(\mathbf{m}_{1:T}^{\mathrm{raw}}, \tau\right), \tag{3}$$

which converts a raw motion clip and terrain height field $h_\tau(x, y)$ into kinematic supervision for teacher training. TCRS is not intended to solve full contact-rich dynamics; instead, it constructs contact-consistent, smooth, and style-preserving references that make terrain adaptation learnable for the downstream policy.

a) Contact-aware terrain reference.: TCRS first estimates stance and swing intervals from foot height, velocity, and hysteresis thresholds. Stance feet are latched to terrain support surfaces, while swing endpoints inherit the raw liftoff and landing timing. This step produces terrain-aware foot targets without changing the global behavior phase of the input clip. The implementation is conservative in horizontal foot placement: it preserves the raw gait phase and nominal footfall timing, uses local support latching and edge penalties around the raw foot trace rather than a global footstep planner, and attenuates or rejects toe/heel constraints when support is unreliable. Thus TCRS should be read as local terrain-conformal reference repair, not as full foothold replanning.

b) Foot-geometry-aware swing optimization.: For each foot f and swing phase s, TCRS optimizes a virtual mid-foot trajectory rather than the ankle origin. Let $\mathbf{r}^{\mathrm{toe}}$ and $\mathbf{r}^{\mathrm{heel}}$ be toe and heel offsets in the ankle frame, and define

$$\mathbf{r}^{\mathrm{mid}} = \frac{1}{2}(\mathbf{r}^{\mathrm{toe}} + \mathbf{r}^{\mathrm{heel}}), \qquad \mathbf{p}_{f,t}^{\mathrm{mid}} = \mathbf{p}_{f,t}^{\mathrm{ankle}} + \mathbf{R}_{f,t}\mathbf{r}^{\mathrm{mid}}. \tag{4}$$

Planning in the mid-foot frame balances toe and heel clearance near terrain discontinuities. For control knots $\mathbf{Y} = \{\mathbf{y}_k\}_{k=1}^{K}$, the swing objective is

$$\begin{aligned} J_s(\mathbf{Y};\tau) = {} & \lambda_{\mathrm{ref}} \sum_k \|\mathbf{y}_k - \mathbf{y}_k^{\mathrm{raw}}\|^2 + \lambda_{\mathrm{sm}} \sum_k \|\Delta^2 \mathbf{y}_k\|^2 \\ & + \lambda_{\mathrm{clr}} \sum_k [h_\tau(x_k, y_k) + \delta - z_k]_+^2 + \lambda_{\mathrm{edge}} \Phi_{\mathrm{edge}}(\mathbf{Y};\tau) \\ & + \lambda_{\mathrm{end}} \|\mathbf{y}_1 - \bar{\mathbf{y}}_1\|^2 + \lambda_{\mathrm{end}} \|\mathbf{y}_K - \bar{\mathbf{y}}_K\|^2. \end{aligned} \tag{5}$$

where $[x]_+ = \max(x, 0)$. The terms preserve the raw swing, encourage smoothness, enforce terrain clearance margin δ, discourage penetration near vertical faces, and keep liftoff/landing endpoints fixed; Φ_{edge} penalizes toe/heel samples whose neighboring terrain queries indicate a vertical height discontinuity within the foot support footprint. We instantiate this optimizer with batched sampling-based trajectory optimization. With perturbations $\epsilon^{(j)}$ and temperature η, the knot update is

$$\mathbf{Y} \leftarrow \mathbf{Y} + \sum_j \frac{\exp(-J_s(\mathbf{Y} + \epsilon^{(j)})/\eta)}{\sum_\ell \exp(-J_s(\mathbf{Y} + \epsilon^{(\ell)})/\eta)} \epsilon^{(j)}. \tag{6}$$

The optimized mid-foot path is then transformed back to ankle, toe, and heel targets for IK.

c) Support-aware root reconstruction.: After foot replanning, TCRS reconstructs root height from the support contacts. For support weights $w_{f,t}$ and adapted foot positions $(x_{f,t}^{\mathrm{tcrs}}, y_{f,t}^{\mathrm{tcrs}})$, a target root height is

$$z_{\mathrm{root},t}^{\star} = \frac{\sum_f w_{f,t}\left[h_\tau(x_{f,t}^{\mathrm{tcrs}}, y_{f,t}^{\mathrm{tcrs}}) + z_{\mathrm{root},t}^{\mathrm{raw}} - z_{f,t}^{\mathrm{raw}}\right]}{\sum_f w_{f,t} + \epsilon}. \tag{7}$$

The value is clamped by leg reachability and smoothed across support transitions. During flight or weak-contact phases, the filter falls back to the raw vertical profile with limited per-frame displacement.

d) Collision repair and multi-point leg IK.: Finally, TCRS repairs lower-leg and foot collisions, attenuates unsupported toe/heel constraints near step edges, and solves a damped support-aware multi-point Jacobian IK problem over the twelve leg joints. Root translation is fixed to the support-aware reconstruction stage above, and root orientation as well as non-leg joints are preserved from the raw reference. The IK residual stacks ankle, toe, and heel point Jacobians for both feet, with support-aware point weights, posture and continuity regularization, penetration penalties, and damped-least-squares regularization. Multiseed fallback and continuity guards reject high-error or discontinuous branches. The full IK objective is given in Appendix A (Eq. (12)). The output is the paired dataset $(\mathbf{q}^{\mathrm{raw}}, \mathbf{q}^{\mathrm{tcrs}}, \tau)$ for teacher training and student distillation.

C. Policy Architecture and Training

A detailed module-level diagram of the deployable policy is given in Appendix G (Figure 7); we summarize the key components here.

a) Blind teacher.: The blind teacher uses a Transformer actor–critic with tokenized proprioceptive history and reference-command windows. It receives TCRS references as commands and is trained with PPO to track adapted anchor, orientation, foot-position, and foot-velocity targets, with energy and lateral foot/shin contact penalties. Appendix A gives the observation and reward contract.

b) Identity-gated vision student.: The student inherits the same command/history Transformer and adds a height-map encoder. Denote the pooled command-history latent by $\mathbf{u}_t$ and the visual latent by $\mathbf{z}_t^{\text{vis}} = E_{\text{vis}}(\mathbf{o}_t^{\text{vis}})$. Terrain affects the actor through two zero-initialized residual pathways. The intent latent is modulated as

$$\mathbf{u}_t' = \mathbf{u}_t + \tanh(\boldsymbol{\alpha}_u) \odot f_u(\mathbf{z}_t^{\text{vis}}), \tag{8}$$

and the action mean is

$$\boldsymbol{\mu}_t = \boldsymbol{\mu}_t^{\text{base}} + \tanh(\boldsymbol{\alpha}_a) \odot f_a([\mathbf{o}_t^{\text{prop}}, \mathbf{z}_t^{\text{vis}}]). \tag{9}$$

Both gate vectors and final residual layers are initialized at zero, so the terrain pathway is inactive at initialization. The student therefore begins as a raw-reference tracker and learns terrain-conditioned corrections only when they improve the tracking objective.

c) Target-frame distillation and PPO fine-tuning.: Because teacher and student act around different command frames, the student cannot imitate the teacher residual directly. We instead distill the teacher's effective PD target, expressed relative to the raw reference:

$$\mathbf{a}_t^\star = \left(\mathbf{q}_t^{\text{tcrs}} + \boldsymbol{\mu}_t^{\text{tea}}\right) - \mathbf{q}_t^{\text{raw}}. \tag{10}$$

The student minimizes $\|\boldsymbol{\mu}_t^{\text{stu}} - \mathbf{a}_t^\star\|_2^2$. During DAgger-style rollouts, the teacher-control probability is annealed from 1 to 0; when the teacher controls the simulator, the applied action is the aligned target-frame action rather than the teacher's native adapted-reference residual. PPO fine-tuning then uses raw-reference commands, height maps, and auxiliary estimator losses, with a lower learning-rate scale on the transferred backbone than on the terrain encoder, critic, and residual branches. The aligned label and student mean are compared in the same residual joint-position units as Eq. (2), after the action scaling used by the shared PD-target convention; the same residual limits and clipping are applied when aligned labels are used for teacher-control rollouts. Although PPO fine-tuning is commanded by the raw reference, it is not learning terrain adaptation from scratch. Target-frame distillation initializes terrain corrections as residual actions around the raw command, and fine-tuning refines those corrections while contact and collision penalties discourage collapse back to flat-ground tracking. The raw tracking terms therefore preserve the user command, while the initialized residual pathway supplies the terrain-aware correction channel.

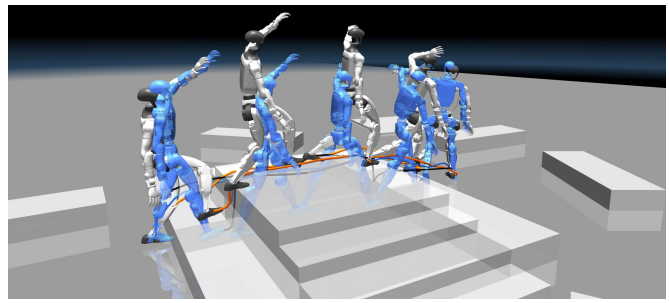

Fig. 3. **TCRS trajectory synthesis.** The blue ghost is the raw reference placed on terrain; the opaque robot is the TCRS output. Foot traces compare the sampling-based (model predictive path integral, MPPI) foot-end optimization used in TCRS (yellow), Cubic Interp (blue), and direct terrain-height z-lifting (black).

IV. Experiments

We evaluate Perceptive BFM at five levels. First, we measure the quality of TCRS outputs before any policy rollout, isolating whether the reference synthesizer produces terrain-conformal supervision (Sec. IV-A). Second, within a common 10k-iteration final-PPO ablation family, we report end-of-training logged rollout diagnostics for the resulting policies (Sec. IV-B). Third, we assess the downstream value of TCRS supervision by comparing reference-source choices (Sec. IV-C). Fourth, we run a cross-terrain rollout evaluation in simulation against the raw-reference baseline (Sec. IV-D). Fifth, we document real-robot deployment of a single policy across diverse behaviors and terrains, including mocap-based operator–environment mismatch (Sec. IV-E). Levels two through four are simulation evidence; level five is real-hardware deployment.

Setup. TCRS supervision is generated from locomotion-oriented human motion clips on procedural terrain with height offsets in $[-0.10, 0.25]$ m, support widths in $[0.10, 1.00]$ m, stair risers 5–15 cm, treads in $[0.20, 0.35]$ m, and slopes up to $30°$. All final-PPO variants use the same simulator, task definition, reward terms, observation contract, action space, environment count, and 10k-iteration budget at 50 Hz control; stage ablations may differ in warm start when the ablated stage is the stage that produces that warm start. For Table II, closed-loop success is computed from TensorBoard counters as $\text{total_success}/(\text{total_success}+\text{total_failed})$. The available log provenance does not establish that this counter is equivalent to 750-step timeout completion or that it shares a denominator with the episode-length scalar, so we report it only as a logged success statistic. These logs do not verify a separate held-out command–terrain split for Table II. Real-robot trials follow Appendix I.

A. Quality of Terrain-Conformal Reference Synthesis

Table I evaluates TCRS on stepping stones with stairs, aggregated over 30 motion clips. The baselines isolate simpler geometry edits: *Z-offset* forward-kinematically lifts ankles to the local terrain height, while *Cubic Interp+IK* smooths a height-projected ankle path with cubic interpolation and solves single-point IK. TCRS instead replans a mid-foot swing

Table I

TCRS reference quality on stepping stones with stairs. TCRS is evaluated before any policy rollout, aggregated over 30 motion clips on the same terrain family. Lower is better for all metrics; bold marks the best per column. TCRS mainly improves collision-sensitive diagnostics, especially clearance violation and penetration depth relative to Z-offset, while preserving upper-body style. It is not uniformly best on every diagnostic: Cubic Interp+IK remains close in penetration and scores better on smoothness and upper-body deviation, while Z-offset scores better on float rate.

Reference	Pen. Depth (cm) ↓	Float Rate (%) ↓	Clear. Viol. (%) ↓	Foot Smooth (m/s^2) ↓	Upper Dev. (cm) ↓
Z-offset (FK projection)	5.48	**12.4**	33.8	15.1	6.51
Cubic Interp + IK	2.69	31.7	14.3	**6.9**	**3.98**
TCRS (ours)	**2.38**	32.3	**7.4**	8.6	4.00

Table II

End-of-training logged rollout diagnostics on stepping stones with stairs. All variants share the task, reward, observation contract, action space, and final PPO budget. Values are mined from the final 100 logged iterations. Success rate is $\mathrm{total_success}/(\mathrm{total_success} + \mathrm{total_failed})$ in those logs; the logs do not verify a separate held-out command–terrain split or expose enough logger-level provenance to equate the success counter with timeout completion. Reward and tracking errors are logged means from the same window and should be interpreted as diagnostics because policies may terminate at different times. Stage ablations that remove the distillation warm start (*w/o TCRS*, *w/o vision*, *w/o target-frame align*) are trained from scratch by construction, so the reward column is not directly comparable across variants and is left unbolded; bold marks the numerically highest or lowest value in the comparable columns.

Variant	Succ. (%) ↑	Reward ↑	Anchor Err. (m) ↓	Joint Err. (rad) ↓
PMT (full)	**55.1**	45.2	**0.159**	**1.243**
w/o identity gate	30.4	14.9	0.231	1.428
vision-concat (no gate)	29.9	14.5	0.234	1.421
w/o TCRS (raw ref.)	27.3	17.9	0.229	1.833
w/o target-frame align	26.7	9.3	0.221	1.959
w/o vision (blind)	26.5	10.6	0.238	2.464

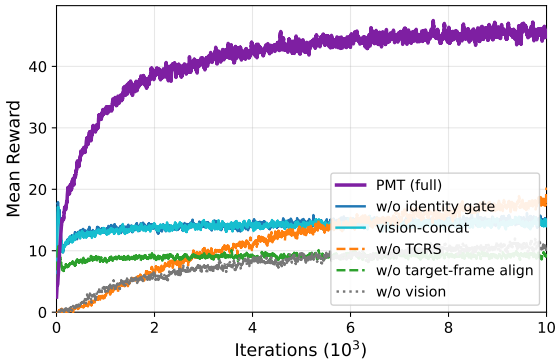

Fig. 4. **Training reward diagnostics.** Mean reward over training for the *same six variants* as Table II, under the common final-PPO task, reward, observation contract, and logging protocol. The full PMT fine-tuning stage resumes from the distilled checkpoint while the distillation-stage ablations train from scratch, so absolute reward is not directly comparable across variants; we use this curve only as a coarse optimization diagnostic and interpret Table II as end-of-training logged rollout diagnostics rather than as a held-out evaluation.

trajectory and reconstructs the body through support-aware root shaping plus multi-point ankle/toe/heel IK, primarily targeting collision-aware swing along the path. Its clearest gains are collision-sensitive: penetration depth drops relative to Z-offset ($5.48 \rightarrow 2.38$ cm), and clearance violation drops relative to Cubic Interp+IK ($14.3 \rightarrow 7.4\%$). TCRS is not uniformly better on every diagnostic: Cubic Interp+IK remains close in penetration and scores better on smoothness and upper-body deviation, while Z-offset scores better on float rate. We therefore interpret Table I as evidence that TCRS improves collision-sensitive reference feasibility, especially clearance, while trading off contact-fidelity and smoothness diagnostics. The higher float rate means some nominal-contact frames sit above terrain according to the stance metric, likely near support transitions or step edges where the synthesizer prioritizes avoiding collision. The downstream question is whether these geometric gains translate into policy performance, which we assess in Sec. IV-C.

B. End-of-Training Logged Rollout Diagnostics

Table II reports the common 10k-iteration ablation family as end-of-training logged rollout diagnostics. Logged closed-loop success rate is the primary metric, with tracking error and reward retained as secondary diagnostics. The full PMT policy reaches 55.1% logged success, above every listed ablation, and attains the lowest anchor error (0.159 m) and the lowest joint error (1.243 rad). Because failed policies can terminate at different times, tracking errors may average over different state distributions; we therefore avoid fine-grained rankings based on small error differences and treat success as the primary diagnostic. We omit the episode-length scalar from the main ablation because the available logs do not establish that it is computed over the same denominator as the success counter.

Perception, TCRS supervision, and target-frame alignment are jointly important. Removing vision, TCRS supervision, or target-frame alignment reduces logged success to around 26–27%. Without seed-level variance, the 0.2–0.8 point differences among these ablations should not be ranked; the coarse observation from these logged runs is that each stage is far below PMT under this setup. The blind row is a stage ablation rather than a pure capacity-matched vision isolation: it removes the terrain encoder and the two identity-gated residual pathways, and it is trained from scratch because the removed perception/distillation path is part of the warm-start pipeline.

TCRS supervision is associated with higher logged success under this training setup. Training the same deployable architecture directly with PPO on raw flat-ground references (*w/o TCRS*) reaches 27.3% logged success—roughly half of PMT—suggesting that terrain-conformal supervision is associated with better terrain adaptation under this training setup, rather than attributing the gain only to perceptual input.

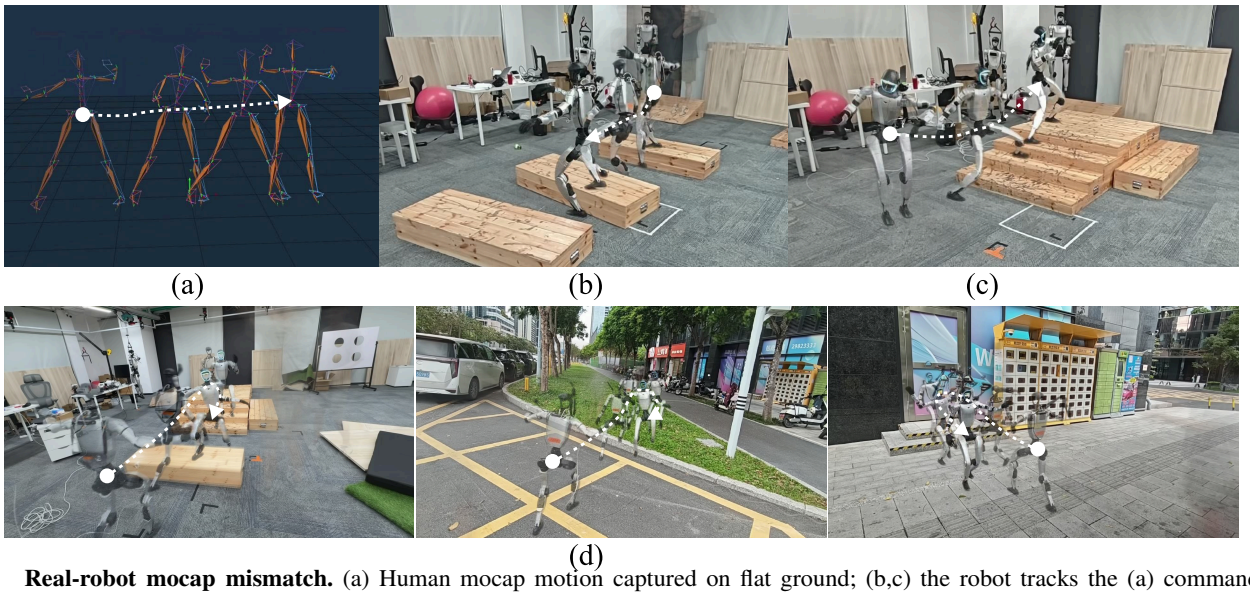

Fig. 5. **Real-robot mocap mismatch.** (a) Human mocap motion captured on flat ground; (b,c) the robot tracks the (a) command over robot-side terrain; (d) a separate walk-and-dance motion deployed in the wild.

Target-frame alignment and identity gating appear beneficial. Removing target-frame distillation (*w/o target-frame align*) yields low logged reward (9.3), elevated joint error (1.959 rad), and 26.7% logged success, consistent with the teacher target being harder to optimize when it is not expressed in the raw-reference action frame. Replacing the zero-initialized identity gate with a plain residual or a vision-concat fusion (*w/o identity gate*, *vision-concat*) reaches about 30% logged success, still below PMT. These rows suggest that initializing as an exact raw-reference tracker and learning terrain corrections as gated residuals is useful, but seed-level variance is needed before ranking the gated variants precisely.

On comparison fairness. The three stage ablations that remove the distillation phase (*w/o TCRS*, *w/o vision*, *w/o target-frame align*) are trained from scratch by construction, because each ablates a stage that contributes to the warm start; the full PMT policy resumes its fine-tuning from the distilled checkpoint. These runs therefore match the final-PPO budget and logging protocol, not the total offline pipeline compute. We treat training reward (Fig. 4) only as a coarse diagnostic and use Table II as end-of-training logged diagnostics for this training setup; a separate held-out command–terrain evaluation remains the stronger validation.

C. Downstream Value of TCRS Supervision

Table III evaluates whether the reference-quality gains of Sec. IV-A translate into better control. The Raw and TCRS rows are the same logged raw-reference and TCRS-supervised runs reported in the component diagnostics, while the Z-offset and Cubic+IK rows evaluate simpler geometric reference sources under the same logged protocol. The reported source-ablation diagnostics show a large gap between raw-reference supervision (27.3% logged success) and TCRS supervision (55.1%), with intermediate gains from the two simpler geometry edits. Without seed-level variance, we read this as evidence from the tested runs rather than a statistical monotonic claim.

Table III

Downstream value of TCRS supervision. Reference-source ablation on stepping stones with stairs. The raw and TCRS endpoints are the same logged runs reported in the component diagnostics. Intermediate source-ablation rows report simpler geometric reference edits under the same logged protocol. These rows are logged source-ablation diagnostics rather than seed-averaged held-out evaluations; without seed-level variance, the ordering should be read as evidence for the tested runs rather than a statistical monotonic claim.

Teacher supervision	Succ. (%) ↑	Anchor Err. (m) ↓	Joint Err. (rad) ↓
Raw reference (no synthesis)	27.3	0.229	1.833
Z-offset references	33.0	0.205	1.61
Cubic Interp + IK references	41.0	0.188	1.44
TCRS references (ours)	**55.1**	**0.159**	**1.243**

D. Cross-Terrain Simulation Rollout Evaluation

Table IV records a separate fixed-policy closed-loop rollout protocol *in simulation*: the evaluation runs 30 motion commands per terrain family across 3 seeds for each fixed policy. These rollout summaries are not TensorBoard counters from Table II. We compare the full Perceptive BFM against the raw-reference policy trained without TCRS supervision and report rollout failure modes (completion, fall, tracking-loss, and lower-leg collision). We deliberately do not report body–terrain penetration here: inside the contact solver, penetration reflects solver tolerance and height-field discretization rather than policy quality, so penetration is a property of the kinematic references (Table I) and lower-leg collision is the geometry-failure diagnostic for closed-loop rollouts. In the aggregate over 450 rollouts, PMT reports higher completion (248/450, 55.1%) than the raw-reference baseline (123/450, 27.3%) and lower lower-leg collision incidence (45/450, 10.0% versus 111/450, 24.7%), while the per-terrain rows report higher completion in each terrain family.

Table IV

Cross-terrain rollout evaluation *in simulation*. The protocol assigns 90 rollout slots to each terrain family (30 motion commands $\times$ 3 seeds) for a fixed policy and a common terrain generator. *Comp.*, *Fall*, and *Track-loss* are mutually exclusive episode outcomes and sum to 100%: *Comp.* reaches the goal horizon without termination, *Fall* is a hard fall, and *Track-loss* is a tracking-divergence termination. *Coll.* is a separate non-terminating incidence rate—the fraction of rollouts with at least one lower-leg/terrain contact—and is therefore not part of the outcome budget. We compare the full Perceptive BFM against the raw-reference, no-TCRS ablation under this separate fixed-policy rollout protocol; these rollout summaries do not reuse TensorBoard counters from Table II. Penetration depth is reported only for kinematic references (Table I); inside the contact solver it reflects solver tolerance rather than policy quality, so we use lower-leg collision as the geometry-failure diagnostic here.

	PMT (ours)				**w/o TCRS (raw ref.)**			
Terrain	Comp.%↑	Fall%↓	Track-loss%↓	Coll.%↓	Comp.%↑	Fall%↓	Track-loss%↓	Coll.%↓
Stairs	53.3	25.6	21.1	11.1	24.4	47.8	27.8	26.7
Slopes	66.7	15.6	17.8	5.6	35.6	37.8	26.7	17.8
Sparse supports	42.2	33.3	24.4	15.6	17.8	55.6	26.7	33.3
Recessed obstacles	52.2	26.7	21.1	10.0	25.6	46.7	27.8	24.4
Mixed indoor	61.1	17.8	21.1	7.8	33.3	41.1	25.6	21.1
All	**55.1**	23.8	21.1	10.0	27.3	45.8	26.9	24.7

E. Real-Robot Deployment Across Behaviors and Terrains

We deploy the same policy weights on a 29-DoF Unitree G1 with onboard depth-to-height-map perception across indoor, outdoor, and mocap-mismatch protocols (Appendix I). Because several behaviors are high-energy and contact-rich, we present these results as qualitative deployment evidence rather than repeated-trial statistics, and rely on the simulation evaluations above for quantitative context. Figure 1 summarizes a *single* raw-reference policy across eight maneuver–terrain pairs: a one-leg backflip and a step-to-stair backflip on raised blocks, a stair dance, an arm-waving run, a free-arm walk over uneven terrain, a sideways stair walk, a backward step over obstacles, and a turning gait on stairs. Each panel pairs a distinct flat-ground command with a different, randomly placed terrain; the policy preserves the commanded upper-body behavior while perception resolves the footholds, clearance, posture, and timing the reference never specifies. Acrobatics, expressive motion, and omnidirectional locomotion are all realized without per-command tuning or a per-terrain controller. Figure 5 isolates the operator–environment mismatch case: the human performs a mocap command on flat ground while the robot executes it over randomly placed terrain.

V. Conclusion

We presented Perceptive BFM, a motion-reference-conditioned humanoid control framework that keeps the raw kinematic reference as the deployment command while grounding its terrain-dependent realization in robot-centric perception. TCRS supplies terrain-conformal supervision offline, PMT transfers that behavior to a raw-reference vision student, and identity-gated residuals let perception correct footholds, clearance, posture, and timing without replacing the command. Our evidence spans quantitative TCRS reference quality, end-of-training logged rollout diagnostics associating TCRS supervision with higher success under the final-PPO setup, cross-terrain simulation rollouts, and qualitative real-robot deployment across diverse behaviors and terrains.

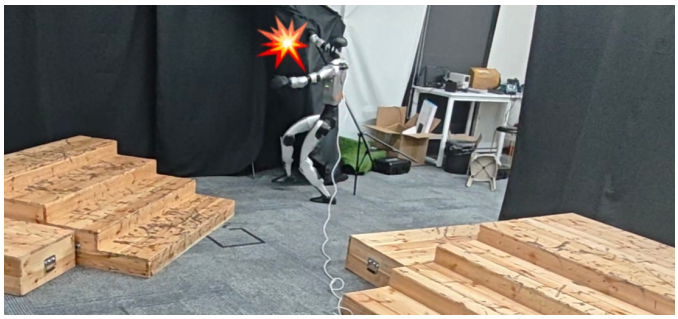

Fig. 6. **Representative failure.** The upper-body command is collision-unaware, so arms or torso can strike obstacles.

VI. Limitations

Assumptions. TCRS is a *kinematic* synthesizer: it builds contact-consistent, style-preserving references without solving contact-rich dynamics, and assumes a static, rigid, observable height field, so it does not model deformable, granular, or slippery media, and assumes the upper-body command stays feasible after lower-body correction.

Evaluation. The main ablation table uses final-window TensorBoard counters rather than a separate held-out command–terrain split with seed-level variance, so we treat it as logged rollout diagnostics. A larger fixed-checkpoint held-out evaluation is the next step for stronger statistical evidence.

Failure modes. Because adaptation is foot-centric while the upper-body command is preserved as-is, the arms and torso can strike nearby obstacles (Figure 6). **Future work**: collision-aware upper-body adaptation, dynamic and deformable terrain, and scaling the behavior corpus toward a fuller foundation-model regime.

References

[1] Ananye Agarwal, Ashish Kumar, Jitendra Malik, and Deepak Pathak. Legged locomotion in challenging terrains using egocentric vision. In *Proceedings of the 6th Conference on Robot Learning*, volume 205 of

Proceedings of Machine Learning Research, pages 403–415. PMLR, 2023. URL https://proceedings.mlr.press/v205/agarwal23a.html.

[2] Xuxin Cheng, Yandong Ji, Junming Chen, Ruihan Yang, Ge Yang, and Xiaolong Wang. Expressive whole-body control for humanoid robots. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, July 2024. doi: 10.15607/RSS.2024.XX.107.

[3] Xuxin Cheng, Kexin Shi, Ananye Agarwal, and Deepak Pathak. Extreme parkour with legged robots. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11443–11450, 2024. doi: 10.1109/ICRA57147.2024.10610200.

[4] Sungjoon Choi, Matthew K. X. J. Pan, and Joohyung Kim. Nonparametric motion retargeting for humanoid robots on shared latent space. In *Robotics: Science and Systems*, 2020. doi: 10.15607/RSS.2020.XVI.071.

[5] William D. Compton, Zachary Olkin, and Aaron D. Ames. Terrain consistent reference-guided RL for humanoid navigation autonomy. *arXiv preprint arXiv:2605.15517*, 2026. doi: 10.48550/arXiv.2605.15517.

[6] Ewen Dantec, Maximilien Naveau, Pierre Fernbach, Nahuel A. Villa, Guilhem Saurel, Olivier Stasse, Michel Taix, and Nicolas Mansard. Whole-body model predictive control for biped locomotion on a torque-controlled humanoid robot. *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pages 638–644, 2022. doi: 10.1109/Humanoids53995.2022.10000129.

[7] Zipeng Fu, Qingqing Zhao, Qi Wu, Gordon Wetzstein, and Chelsea Finn. HumanPlus: Humanoid shadowing and imitation from humans. In *Proceedings of the 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pages 2828–2844. PMLR, 2025. URL https://proceedings.mlr.press/v270/fu25a.html.

[8] 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)*, pages 8944–8951, 2024. doi: 10.1109/IROS58592.2024.10801984.

[9] Tairan He, Wenli Xiao, Toru Lin, Zhengyi Luo, Zhenjia Xu, Zhenyu Jiang, Jan Kautz, Changliu Liu, Guanya Shi, Xiaolong Wang, Linxi Fan, and Yuke Zhu. HOVER: Versatile neural whole-body controller for humanoid robots. *arXiv preprint arXiv:2410.21229*, 2024. doi: 10.48550/arXiv.2410.21229.

[10] Tairan He, Jiawei Gao, Wenli Xiao, Yuanhang Zhang, Zi Wang, Jiashun Wang, Zhengyi Luo, Guanqi He, Nikhil Sobanbab, Chaoyi Pan, Zeji Yi, Guannan Qu, Kris Kitani, Jessica Hodgins, Linxi "Jim" Fan, Yuke Zhu, Changliu Liu, and Guanya Shi. ASAP: Aligning simulation and real-world physics for learning agile humanoid whole-body skills. *arXiv preprint arXiv:2502.01143*, 2025. doi: 10.48550/arXiv.2502.01143.

[11] Tairan He, Zhengyi Luo, Xialin He, Wenli Xiao, Chong Zhang, Weinan Zhang, Kris M. Kitani, Changliu Liu, and Guanya Shi. OmniH2O: Universal and dexterous human-to-humanoid whole-body teleoperation and learning. In *Proceedings of the 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pages 1516–1540. PMLR, 2025. URL https://proceedings.mlr.press/v270/he25b.html.

[12] Tobias Johannink, Shikhar Bahl, Ashvin Nair, Jianlan Luo, Avinash Kumar, Matthias Loskyll, Juan Aparicio Ojea, Eugen Solowjow, and Sergey Levine. Residual reinforcement learning for robot control. *IEEE International Conference on Robotics and Automation (ICRA)*, pages 6023–6029, 2019. doi: 10.1109/ICRA.2019.8794127.

[13] Ashish Kumar, Zipeng Fu, Deepak Pathak, and Jitendra Malik. RMA: Rapid motor adaptation for legged robots. In *Robotics: Science and Systems*, 2021. doi: 10.15607/RSS.2021.XVII.011.

[14] Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning quadrupedal locomotion over challenging terrain. *Science Robotics*, 5(47): eabc5986, 2020. doi: 10.1126/scirobotics.abc5986.

[15] Yitang Li, Zhengyi Luo, Tonghe Zhang, Cunxi Dai, Anssi Kanervisto, Andrea Tirinzoni, Haoyang Weng, Kris Kitani, Mateusz Guzek, Ahmed Touati, Alessandro Lazaric, Matteo Pirotta, and Guanya Shi. BFM-Zero: A promptable behavioral foundation model for humanoid control using unsupervised reinforcement learning. *arXiv preprint arXiv:2511.04131*, 2025. doi: 10.48550/arXiv.2511.04131.

[16] Yuhan Li, Peiyuan Zhi, Yunshen Wang, Tengyu Liu, Sixu Yan, Wenyu Liu, Xinggang Wang, Baoxiong Jia, and Siyuan Huang. OmniTrack: General motion tracking via physics-consistent reference. *arXiv preprint arXiv:2602.23832*, 2026. doi: 10.48550/arXiv.2602.23832.

[17] Zhengyi Luo, Jinkun Cao, Alexander Winkler, Kris Kitani, and Weipeng Xu. Perpetual humanoid control for real-time simulated avatars. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10895–10904, 2023. doi: 10.1109/ICCV51070.2023.01000.

[18] Zhengyi Luo, Jinkun Cao, Josh Merel, Alexander Winkler, Jing Huang, Kris Kitani, and Weipeng Xu. Universal humanoid motion representations for physics-based control. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=OrOd8PxOO2.

[19] Zhengyi Luo, Ye Yuan, Tingwu Wang, Chenran Li, Sirui Chen, Fernando Castaneda, Zi-Ang Cao, Jiefeng Li, David Minor, Qingwei Ben, Xingye Da, Linxi Fan, and Yuke Zhu. SONIC: Supersizing motion tracking for natural humanoid whole-body control. *arXiv preprint arXiv:2511.07820*, 2025. doi: 10.48550/arXiv.2511.07820.

[20] Yubiao Ma, Han Yu, Jiayin Xie, Changtai Lv, Qiang Luo, Chi Zhang, Yunpeng Yin, Boyang Xing, Xuemei Ren, and Dongdong Zheng. Robust and generalized humanoid

motion tracking. *arXiv preprint arXiv:2601.23080*, 2026. doi: 10.48550/arXiv.2601.23080.

[21] Takahiro Miki, Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning robust perceptive locomotion for quadrupedal robots in the wild. *Science Robotics*, 7(62):eabk2822, 2022. doi: 10.1126/scirobotics.abk2822.

[22] NVIDIA, :, Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi "Jim" Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, Joel Jang, Zhenyu Jiang, Jan Kautz, Kaushil Kundalia, Lawrence Lao, Zhiqi Li, Zongyu Lin, Kevin Lin, Guilin Liu, Edith Llontop, Loic Magne, Ajay Mandlekar, Avnish Narayan, Soroush Nasiriany, Scott Reed, You Liang Tan, Guanzhi Wang, Zu Wang, Jing Wang, Qi Wang, Jiannan Xiang, Yuqi Xie, Yinzhen Xu, Zhenjia Xu, Seonghyeon Ye, Zhiding Yu, Ao Zhang, Hao Zhang, Yizhou Zhao, Ruijie Zheng, and Yuke Zhu. Gr00t n1: An open foundation model for generalist humanoid robots, 2025. URL https://arxiv.org/abs/2503.14734.

[23] Adrien Pajon, Stéphane Caron, Giovanni De Magistris, Sylvain Miossec, and Abderrahmane Kheddar. Walking on gravel with soft soles using linear inverted pendulum tracking and reaction force distribution. In *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, pages 432–437, 2017. doi: 10.1109/HUMANOIDS.2017.8246909.

[24] Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel van de Panne. Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. *ACM Transactions on Graphics*, 37(4):143, 2018. doi: 10.1145/3197517.3201311.

[25] Xue Bin Peng, Ze Ma, Pieter Abbeel, Sergey Levine, and Angjoo Kanazawa. AMP: Adversarial motion priors for stylized physics-based character control. *ACM Transactions on Graphics*, 40(4):1–20, 2021. doi: 10.1145/3450626.3459670.

[26] Tom Silver, Kelsey Allen, Josh Tenenbaum, and Leslie Pack Kaelbling. Residual policy learning. *arXiv preprint arXiv:1812.06298*, 2018. doi: 10.48550/arXiv.1812.06298.

[27] Wandong Sun, Yongbo Su, Leoric Huang, Alex Zhang, Dwyane Wei, Mu San, Daniel Tian, Ellie Cao, Finn Yan, Ethan Xie, and Zongwu Xie. Now you see that: Learning end-to-end humanoid locomotion from raw pixels. *arXiv preprint arXiv:2602.06382*, 2026. doi: 10.48550/arXiv.2602.06382.

[28] Ruben Villegas, Jimei Yang, Duygu Ceylan, and Honglak Lee. Neural kinematic networks for unsupervised motion retargetting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8639–8648, 2018. doi: 10.1109/CVPR.2018.00901.

[29] Huayi Wang, Zirui Wang, Junli Ren, Qingwei Ben, Tao Huang, Weinan Zhang, and Jiangmiao Pang. BeamDojo: Learning agile humanoid locomotion on sparse footholds. *arXiv preprint arXiv:2502.10363*, 2025. doi: 10.48550/arXiv.2502.10363.

[30] Yunshen Wang, Shaohang Zhu, Peiyuan Zhi, Yuhan Li, Jiaxin Li, Yong-Lu Li, Yuchen Xiao, Xingxing Wang, Baoxiong Jia, and Siyuan Huang. OmniXtreme: Breaking the generality barrier in high-dynamic humanoid control. *arXiv preprint arXiv:2602.23843*, 2026. doi: 10.48550/arXiv.2602.23843.

[31] Zifan Wang, Yufei Jia, Lu Shi, Haoyu Wang, Haizhou Zhao, Xueyang Li, Jinni Zhou, Jun Ma, and Guyue Zhou. Arm-constrained curriculum learning for loco-manipulation of a wheel-legged robot. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10770–10776. IEEE, 2024. doi: 10.1109/IROS58592.2024.10802062.

[32] Zifan Wang, Teli Ma, Yufei Jia, Xun Yang, Jiaming Zhou, Wenlong Ouyang, Qiang Zhang, and Junwei Liang. Omni-perception: Omnidirectional collision avoidance for legged locomotion in dynamic environments. *arXiv preprint arXiv:2505.19214*, 2025. doi: 10.48550/arXiv.2505.19214.

[33] Zifan Wang, Xun Yang, Jianzhuang Zhao, Jiaming Zhou, Teli Ma, Ziyao Gao, Arash Ajoudani, and Junwei Liang. End-to-end humanoid robot safe and comfortable locomotion policy. *arXiv preprint arXiv:2508.07611*, 2025. doi: 10.48550/arXiv.2508.07611.

[34] Zhen Wu, Xiaoyu Huang, Lujie Yang, Yuanhang Zhang, Koushil Sreenath, Xi Chen, Pieter Abbeel, Rocky Duan, Angjoo Kanazawa, Carmelo Sferrazza, Guanya Shi, and C. Karen Liu. Perceptive humanoid parkour: Chaining dynamic human skills via motion matching. *arXiv preprint arXiv:2602.15827*, 2026. doi: 10.48550/arXiv.2602.15827.

[35] Lujie Yang, Xiaoyu Huang, Zhen Wu, Angjoo Kanazawa, Pieter Abbeel, Carmelo Sferrazza, C. Karen Liu, Rocky Duan, and Guanya Shi. OmniRetarget: Interaction-preserving data generation for humanoid whole-body loco-manipulation and scene interaction. *arXiv preprint arXiv:2509.26633*, 2025. doi: 10.48550/arXiv.2509.26633.

[36] 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. doi: 10.48550/arXiv.2509.13780.

[37] Zewei Zhang, Kehan Wen, Michael Xu, Junzhe He, Chenhao Li, Takahiro Miki, Clemens Schwarke, Chong Zhang, Xue Bin Peng, and Marco Hutter. Learning whole-body humanoid locomotion via motion generation and motion tracking. *arXiv preprint arXiv:2604.17335*, 2026. doi: 10.48550/arXiv.2604.17335.

[38] Siheng Zhao, Yanjie Ze, Yue Wang, C. Karen Liu, Pieter Abbeel, Guanya Shi, and Rocky Duan. ResMimic: From general motion tracking to humanoid whole-body loco-manipulation via residual learning. *arXiv preprint arXiv:2510.05070*, 2025. doi: 10.48550/arXiv.2510.05070.

[39] 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*, 2026. doi: 10.48550/arXiv.2601.07718.

[40] Ziwen Zhuang, Zipeng Fu, Jianren Wang, Christopher G. Atkeson, Sören Schwertfeger, Chelsea Finn, and Hang Zhao. Robot parkour learning. In *Proceedings of the 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pages 73–92. PMLR, 2023. URL https://proceedings.mlr.press/v229/zhuang23a.html.

[41] Ziwen Zhuang, Shenzhe Yao, and Hang Zhao. Humanoid parkour learning. *arXiv preprint arXiv:2406.10759*, 2024. doi: 10.48550/arXiv.2406.10759.

[42] Ziwen Zhuang, Shaoting Zhu, Mengjie Zhao, and Hang Zhao. Deep whole-body parkour. *arXiv preprint arXiv:2601.07701*, 2026. doi: 10.48550/arXiv.2601.07701.

APPENDIX

A. PMT Training Algorithm

Algorithm 1 Perceptive Motion Tracking (PMT) training algorithm.

Input: raw motion clips $\mathcal{M}$, terrain generator $\mathcal{G}$, robot model, height scanner
for clip $m \in \mathcal{M}$ and terrain $\tau \sim \mathcal{G}$ **do**
  synthesize $\mathbf{m}^{\mathrm{tcrs}} = \mathcal{S}_{\mathrm{TCRS}}(m, \tau)$ using Alg. 2
  export paired raw/adapted references $(\mathbf{m}^{\mathrm{raw}}, \mathbf{m}^{\mathrm{tcrs}}, \tau)$
end for
train blind teacher π_T with PPO to track $\mathbf{m}^{\mathrm{tcrs}}$
compute target-frame labels from π_T using Eq. (10)
distill identity-gated vision student π_S using raw-reference observations
fine-tune π_S with PPO under raw-reference commands and height-map observations
Output: deployable Perceptive BFM policy π_S

B. Training Stages

Table V maps each of the four PMT stages to its role in the pipeline.

Table V
PMT stage map. Each stage in the PMT training algorithm corresponds to a distinct role in the pipeline.

Stage	Role
TCRS data generation	Synthesizes paired raw/TCRS references on sampled terrain.
Blind teacher PPO	Tracks terrain-conformal references with privileged training groups.
Raw-reference distillation	Distills adapted-reference teacher targets into the raw-reference student frame.
Vision PPO fine-tuning	Fine-tunes the deployable vision student with raw commands and height scans.

C. Observation Contract

D. Reward and Optimization Details

$$r_t = \sum_{k \in \{a,R,f,v\}} w_k \, e^{-\|\Delta_k\|^2/\sigma_k^2} - c_E E_t - c_C C_t, \qquad (11)$$

The PPO reward used by the adapted-reference teacher and the fine-tuning stage combines four exponential tracking terms with two penalties (Eq. (11)). The tracking residuals are anchor position $\Delta_a = \mathbf{p}_t^A - \bar{\mathbf{p}}_t^A$, body orientation $\Delta_R = d_R(\mathbf{R}_t, \bar{\mathbf{R}}_t)$, and ankle position/velocity Δ_f, Δ_v on F. E_t is action/torque energy and C_t is lateral foot/shin contact. Teacher rewards are evaluated against TCRS references, while student fine-tuning uses the raw-reference command frame with terrain-conditioned residual actions. This raw-reference fine-tuning preserves the user command but starts from target-frame distillation, so it refines existing terrain-aware residuals rather than discovering them from scratch; weights and standard deviations are listed in Table VII.

Per-stage learning rates, entropy coefficients, and auxiliary-loss schedules are listed in Table VIII.

E. Network and Data-Flow Summary

a) Architecture.: The actor consumes the raw reference command tokens and a 10-step proprioceptive history through the Transformer backbone, producing the motion-tracking latent. The 17×11 height map together with its validity mask is processed by the terrain encoder into a terrain latent. Two zero-initialized residual branches inject this latent into the actor: an intent gate that modulates the pooled command-history latent (Eq. 8) and an action-residual branch that adds to the action mean (Eq. 9). The actor output is a residual joint-position target around the raw reference, applied through the convention in Eq. (2). The actor also reads two learned estimator heads driven by internal latents only (no privileged observation input): a base-velocity estimator and a motion-anchor-position estimator whose (detached) 3D outputs are concatenated into the actor trunk, so the deployable actor never consumes ground-truth base velocity or anchor position. These estimators are supervised by the corresponding ground-truth targets (base velocity and robot-frame anchor position), which serve as auxiliary losses only; a foot-trajectory head is likewise supervised against the teacher target and is not an actor input. The future motion-anchor displacement window, in contrast, remains a direct command-token input (concatenated with the reference-command tokens), not an estimated quantity. Figure 2 in the main paper illustrates the corresponding overview pipeline.

b) Data flow.: Each raw motion clip is paired with a sampled terrain to form (m^{raw}, τ), which TCRS converts into a terrain-conformal reference m^{tcrs} (Algorithm 2). The blind teacher is trained with PPO on m^{tcrs}. During distillation, the teacher's action is converted into the raw-reference frame using the target-frame relabeling in Eq. (10), and the raw-reference student is fit by MSE. The converted action uses the same scaled residual joint-position convention as the student action and is clipped by the same residual limits before it is used for imitation or teacher-control rollouts. The student is then fine-tuned with PPO under raw-reference commands and onboard height-scan observations. At deployment, the policy receives only the raw reference, proprioception, and the onboard terrain observation; TCRS supervision is never queried online.

F. Terrain-Conformal Reference Synthesis Details

a) Multi-point leg IK objective.: Each frame, TCRS solves a damped support-aware multi-point Jacobian IK problem over the twelve leg joints; root translation is fixed to the support-aware reconstruction stage and root orientation and non-leg joints are preserved from the raw reference. For foot point set $\mathcal{P} = \{\mathrm{ankle}, \mathrm{toe}, \mathrm{heel}\}$, the local IK update solves

$$\begin{aligned}\Delta\mathbf{q}^{\mathrm{leg},\star} = \arg\min_{\Delta\mathbf{q}^{\mathrm{leg}}} & \sum_{f\in\{L,R\}} \sum_{p\in\mathcal{P}} \left\| \mathbf{W}_{f,p} \left(\mathbf{J}_{f,p} \Delta\mathbf{q}^{\mathrm{leg}} - \mathbf{e}_{f,p} \right) \right\|^2 \\ & + \lambda_{\mathrm{post}} \|\mathbf{q}^{\mathrm{leg}} - \mathbf{q}^{\mathrm{raw,leg}}\|^2 \\ & + \lambda_{\mathrm{cont}} \|\mathbf{q}^{\mathrm{leg}} - \mathbf{q}^{\mathrm{prev,leg}}\|^2 \\ & + \lambda_{\mathrm{pen}} \Psi_{\mathrm{pen}}(\mathbf{q}^{\mathrm{leg}}) + \lambda_{\mathrm{dls}} \|\Delta\mathbf{q}^{\mathrm{leg}}\|^2, \end{aligned} \qquad (12)$$

where $\Delta\mathbf{q}^{\mathrm{leg}}$ contains the twelve leg-joint increments, $\mathbf{e}_{f,p}$ are ankle/toe/heel residuals, $\mathbf{W}_{f,p}$ are support-aware point weights,

Table VI
Observation groups used by PMT. Privileged groups supervise critics or auxiliary heads but are not direct deployable actor inputs.

Group	Main contents	Shape or history	Deployment role
Proprio	projected gravity, base angular velocity, joint pos/vel, previous action	93D	actor input
Proprio history	unflattened proprioceptive history	10 steps	actor temporal input
Command window	future reference velocity, gravity, joint command tokens	21×38	actor command input
Anchor delta window	local anchor displacement window	21×3	actor command input
Vision	height scan plus validity mask	17×11 cells, mask appended	vision actor input
Critic	privileged reference/body/base information	current frame	training only
Auxiliary targets	base velocity, anchor, and foot-trajectory targets	current frame/window	auxiliary losses

Table VII
PPO reward terms for the teacher and fine-tuning tasks. Exponential tracking terms use the listed standard deviation.

Term	Weight	Notes
Global anchor position tracking	1.0	std 0.2 m
Relative body orientation tracking	0.5	std 0.35
Foot position tracking	1.0	ankle bodies, std 0.1 m
Foot linear-velocity tracking	0.5	ankle bodies, std 1.0 m/s
Energy	-2×10^{-5}	action/torque energy penalty
Foot/shin lateral contact	-0.03	threshold 5 N on ankles and knees

and Ψ_{pen} penalizes foot or lower-leg penetration. Multiseed fallback and continuity guards reject high-error or discontinuous branches.

Algorithm 2 Terrain-Conformal Reference Synthesis (TCRS).

Input: raw clip m^{raw}, terrain height field h_τ, contact offsets, swing-optimization parameters, IK weights
detect contact masks using foot height, speed, and hysteresis thresholds
build stance references by anchoring support feet to terrain surfaces
for each foot f and swing phase $s = [t_0, t_1]$ **do**
  convert ankle targets to a mid-foot frame using Eq. (4)
  initialize trajectory knots from the raw mid-foot path
  for sampling iteration $i = 1, \ldots, N_{\text{iter}}$ **do**
    sample temporally smoothed knot perturbations
    evaluate tracking, smoothness, clearance, vertical-face, and endpoint costs using Eq. (5)
    update knots with the softmin-weighted sampling update in Eq. (6)
  end for
  convert optimized mid-foot trajectory back to ankle, toe, and heel targets
end for
reconstruct root height from support contacts using Eq. (7)
repair lower-leg and foot collisions; attenuate unsupported toe/heel point weights near edges
for frame $t = 1, \ldots, T$ **do**
  solve multi-point Jacobian IK using Eq. (12)
  retry with secondary seeds if foot error, penetration, or joint jumps exceed thresholds
end for
Output: terrain-conformal reference m^{tcrs}, contact masks, realized feet/root, and diagnostics

b) Reference-quality metrics.: We evaluate TCRS before policy rollout, following the principle that a reference generator should be tested independently from the tracker it trains. Let

$$d_\tau(\mathbf{p}) = p_z - h_\tau(p_x, p_y) \tag{13}$$

be signed terrain clearance. Terrain penetration is

$$E_{\text{pen}} = \frac{1}{T|\mathcal{B}|} \sum_{t=1}^{T} \sum_{b\in\mathcal{B}} [-d_\tau(\mathbf{p}_{b,t})]_+, \tag{14}$$

where $\mathcal{B}$ contains foot and lower-leg points. Stance contact error and floating rate are

$$\begin{aligned} E_{\text{contact}} &= \frac{1}{|\mathcal{C}|} \sum_{(f,t)\in\mathcal{C}} |d_\tau(\mathbf{p}_{f,t})|, \\ R_{\text{float}} &= \frac{1}{|\mathcal{C}|} \sum_{(f,t)\in\mathcal{C}} \mathbb{I}[d_\tau(\mathbf{p}_{f,t}) > \epsilon_{\text{float}}]. \end{aligned} \tag{15}$$

Swing-clearance violation is

$$R_{\text{clear}} = \frac{1}{|\mathcal{S}|} \sum_{(f,t)\in\mathcal{S}} \mathbb{I}[d_\tau(\mathbf{p}_{f,t}) < c_{\min}], \tag{16}$$

and style deviation is measured by upper-body deviation from the raw clip,

$$E_{\text{upper}} = \frac{1}{T|\mathcal{J}_{ub}|} \sum_{t,j\in\mathcal{J}_{ub}} \|\mathbf{x}_{j,t}^{\text{tcrs}} - \mathbf{x}_{j,t}^{\text{raw}}\|_2. \tag{17}$$

Foot smoothness is measured as mean finite-difference foot acceleration,

$$E_{\text{smooth}} = \frac{1}{(T-2)|\mathcal{F}|} \sum_{t=2}^{T-1} \sum_{f\in\mathcal{F}} \left\| \frac{\mathbf{p}_{f,t+1} - 2\mathbf{p}_{f,t} + \mathbf{p}_{f,t-1}}{\Delta t^2} \right\|_2. \tag{18}$$

A good synthesizer should reduce penetration and clearance violations, avoid excessive nominal-contact floating, keep E_{upper} small, and maintain smooth foot trajectories. The corresponding values are reported in the main paper as Table I.

c) Implementation specifics.: In our implementation, TCRS preserves the root orientation and upper-body joint references from the raw clip. Root translation is shaped by the support-aware reconstruction stage (Eq. (7)) and is then held fixed during IK, so the IK solver only updates the twelve leg joints through a multi-point Jacobian system over ankle, toe, and heel targets for both feet. The multi-point structure is what enables foot-geometry-aware contact near step edges. This decomposition makes terrain adaptation primarily a lower-body contact adjustment while limiting distortion of the commanded whole-body style; the upper-body trajectory is therefore preserved across both synthesis and deployment.

Table VIII

Training hyperparameters. The fine-tuning stage starts from the distilled vision checkpoint and updates the inherited tracker more conservatively than the terrain branch.

Stage	Learning rate	Entropy	Additional losses or schedule
Blind teacher PPO	5×10^{-4}	0.005	5 epochs, 4 minibatches, KL target 0.01, velocity/anchor Huber losses
Distillation	1×10^{-4}	–	MSE action loss, teacher-control mix annealed $1.0 \rightarrow 0.0$
Vision PPO fine-tuning	1×10^{-4}	0.001	backbone LR scale 0.3, foot-trajectory Huber loss with delta 0.05

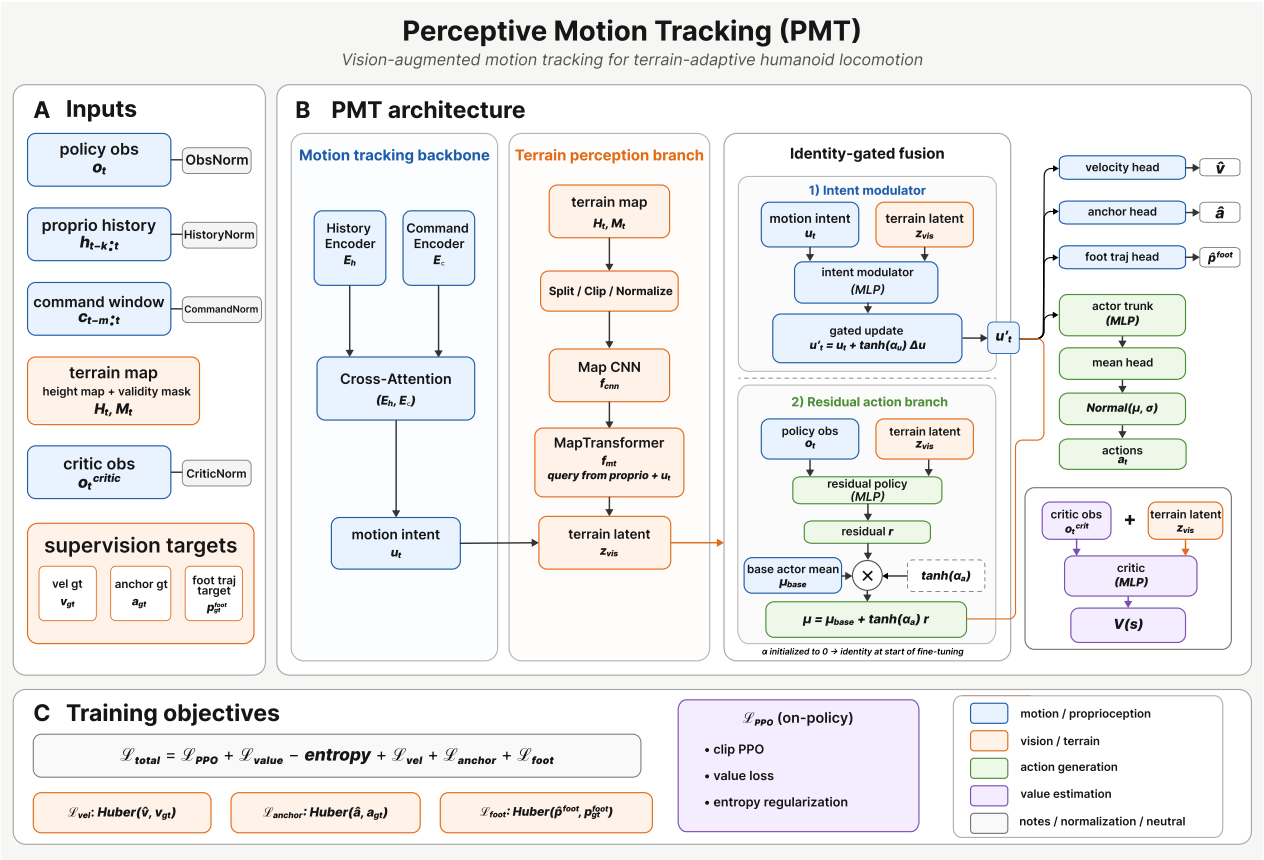

Fig. 7. **Detailed PMT network architecture.** (A) Inputs: policy observation $\mathbf{o}_t$, 10-step proprioceptive history $\mathbf{h}_{t-k:t}$, command window $\mathbf{c}_{t-m:t}$, terrain map $(\mathbf{H}_t, \mathbf{M}_t)$, critic observation, and supervision targets. (B) PMT actor: a Transformer motion-tracking backbone with cross-attention encoders E_h, E_c produces a motion intent $\mathbf{u}_t$; the terrain perception branch (Map CNN f_{cnn} followed by a query-conditioned MapTransformer f_{mt}) produces a terrain latent $\mathbf{z}^{\text{vis}}$; the identity-gated fusion module updates the intent through $\mathbf{u}'_t = \mathbf{u}_t + \tanh(\alpha_u)\Delta\mathbf{u}$ and adds a residual to the action mean $\boldsymbol{\mu} = \boldsymbol{\mu}^{\text{base}} + \tanh(\alpha_a)\, r$, with α initialized at zero so the policy starts as a pure raw-reference tracker. Auxiliary heads predict base velocity, motion anchor, and foot trajectory; the critic operates on privileged inputs. (C) Training objective: PPO + value + entropy losses combined with Huber auxiliary losses on velocity, anchor, and foot trajectory. The corresponding text definitions are in Section III.

G. Network Architecture

H. Training Corpus and Domain Randomization

a) Motion corpus.: TCRS supervision is generated from a locomotion- and dance-oriented human motion-capture corpus (walking, running, turning, side-stepping, and expressive walk–dance sequences from multiple subjects). Each base clip is geometrically augmented by random planar placement and yaw on sampled terrain, producing on the order of 8000 paired raw/terrain-conformal training trajectories. Each motion is exported at 30 fps and paired with a sampled height field as described in Sec. III-B.

b) Control and simulation.: Training uses IsaacLab/IsaacSim with a physics step of $\Delta t = 5$ ms and control decimation 4, giving a 50 Hz policy rate; the configured episode horizon is 15 s, or 750 policy steps. We do not use the episode-length scalar in the main ablation because the available logs do not establish that it shares a denominator with the success counter. Each variant runs 6144 environments per GPU across 48 A800 GPUs.

c) Domain randomization.: To support sim-to-real transfer we randomize, per episode, the parameters in Table IX. Terrain perception is corrupted by an additive per-cell height-scan noise and a planar drift to emulate depth-to-height-map

error, and the robot is periodically pushed by direct base-velocity perturbation.

Table IX
Domain randomization ranges used during training. Values are sampled per episode unless noted. Operation "add" denotes an additive offset to the nominal value.

Parameter	Range	Notes
Static friction	$[0.3, 1.6]$	per-body material
Dynamic friction	$[0.3, 1.2]$	per-body material
Base CoM offset	± 0.025 m	per axis
Default joint position	± 0.1 rad	reset perturbation, prob. 0.2
Push velocity	± 0.2 m/s	periodic base push
Height-scan noise	± 0.06 m	additive, per cell, clipped

I. Real-Robot Deployment Protocol

a) Platform and perception.: The deployable policy runs on a 29-DoF Unitree G1 humanoid. Robot-centric terrain perception is implemented as a torso-mounted depth-to-height-map pipeline that produces a 17×11 height grid over a 1.6 m $\times$ 1.0 m footprint at 0.1 m resolution. The same height-map representation is used in simulation training and during real-robot deployment so that the observation contract is preserved across the sim-to-real transfer.

b) Indoor protocol.: Indoor trials place rectangular obstacles, raised steps, artificial-turf surfaces, and mixed obstacle layouts in a motion-capture–equipped lab. Each terrain configuration is exercised with multiple commanded behaviors drawn from the broad behavior set, including walking, running, side-walking, and gesture-rich motions. These rollouts are illustrated in the main-paper deployment figures.

c) Outdoor protocol.: Outdoor trials cover stairs, grass, isolated steps, recessed flower beds, and sidewalk transitions. Hardware safety is supervised by a runtime watchdog that triggers a soft fall-over recovery if the estimator detects a torque saturation or a base-orientation excursion outside the training-time envelope.

d) Mocap mismatch protocol.: The mocap mismatch setup deliberately separates the human and robot environments: the human performs the commanded motion on flat ground while the robot executes the corresponding kinematic command over randomly placed steps and cube obstacles. This protocol directly tests whether robot-centric terrain perception can ground a human command that contains no matching terrain information.

e) Nature of the evidence.: The same deployed policy weights are used across all behaviors and terrains, with no per-command reward tuning or per-terrain controller switching. Because several behaviors are high-energy and contact-rich (e.g. backflips on raised blocks), running large repeated-trial campaigns on hardware poses a safety risk to the robot and environment; we therefore present real-robot results as qualitative deployment evidence (Figures 1 and 5) and use the simulation summaries and rollout evaluations in Tables II–IV for quantitative context.

J. Training and Compute Details

All policy variants in Figure 4 and the Section IV-B ablation share the same final-PPO task definition, reward terms (Eq. (11)), observation contract (Table VI), action space, episode horizon, environment count, and PPO hyperparameters. All variants are trained on 48 NVIDIA A800 GPUs for 10k iterations in their final PPO stage, but the full PMT method additionally uses TCRS data generation, adapted-reference teacher training, and raw-reference distillation before fine-tuning. *w/o vision* (BlindRaw) removes the height-map encoder and the two identity-gated residual pathways and is trained from scratch as a stage ablation; it should not be interpreted as a pure capacity-matched vision-only ablation. *w/o TCRS* trains the deployable architecture directly with PPO on raw flat-ground references; *w/o target-frame align* removes the target-frame relabeling so the student distills the teacher's native residual; *w/o identity gate* and *vision-concat* replace the zero-initialized gates with a plain residual and a feature-concatenation fusion, respectively.

a) On the warm start and reward as a diagnostic.: The full PMT fine-tuning stage resumes from its distilled checkpoint, while the three stage ablations that remove the distillation phase (*w/o TCRS*, *w/o vision*, *w/o target-frame align*) are trained from scratch, because each ablates a stage that contributes to the warm start. This makes the two-segment PMT reward curve in Figure 4 not directly comparable across variants in absolute reward, and it also means the full pipeline has additional offline compute not counted in the final-PPO budget. We therefore treat training reward only as a coarse optimization diagnostic and base the reported ablation discussion on the logged closed-loop success rate and tracking error in Table II.

b) Closed-loop metrics.: For Table II, closed-loop success rate is mined from the 2026-05-28 TensorBoard ablation family as $\mathrm{total_success}/(\mathrm{total_success}+\mathrm{total_failed})$ averaged over the final 100 logged iterations. The available notes identify this ratio but do not expose enough logger-level provenance to equate $\mathrm{total_success}$ with 750-step timeout completion or to align the episode-length scalar with the same denominator. Anchor error, joint error, and reward are logged end-of-training diagnostics from the same window; the logs do not verify a separate held-out command–terrain split for this table. We therefore treat Table II as logged rollout diagnostics rather than a held-out or total-compute-matched evaluation. Table IV reports a separate fixed-policy cross-terrain rollout protocol of 30 commands per terrain family and 3 seeds. These rollout summaries are not TensorBoard counters from Table II; failure-mode and collision entries summarize the same rollout protocol.