NeCR: Neural Contact-Consistent Retargeting for Whole-Body Humanoid Tracking

Anonymous authors
Paper under double-blind review

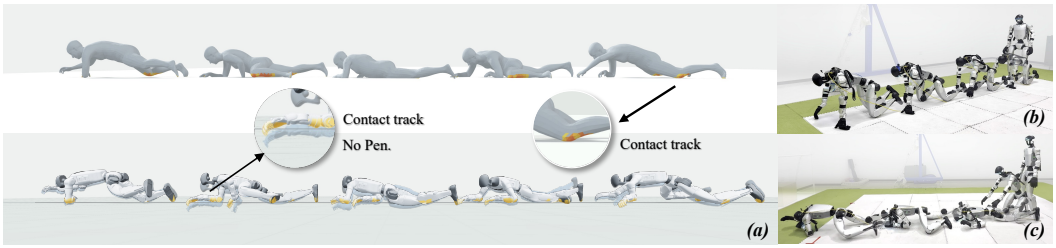

Figure 1: NeCR retargets human motion into contact-consistent robot references, and proprioceptive tracking policies trained on them transfer zero-shot to a Unitree G1. **(a)** A crawl projected onto a penetration-free trajectory: the forearm and knee contacts of the source are reproduced on the robot (insets). **(b, c)** The resulting policies on hardware.

Abstract

Retargeting human motion to a humanoid becomes particularly challenging for multi-contact motions, in which the knees, shins, and forearms support the body alongside the feet. Crawling, kneeling, and getting up from the floor depend on such whole-body contacts, and a robot with different proportions may be unable to reproduce contacts as the source human made them. Learning-based retargeters preserve motion semantics but cannot guarantee physical validity at inference, so a contact may be reproduced while penetrating the ground or sliding. Optimization-based retargeters enforce physical constraints, either frame by frame, where they can misplace support, or over the whole sequence, at a cost that grows with its length. To address this challenge, we present **NeCR**, a hybrid neural–optimization framework in which a learned contact model specifies where, when, and how strongly the optimizer corrects the motion. A neural retargeter first produces a semantically faithful nominal trajectory. Then a temporal model predicts continuous whole-body contact probabilities for the optimizer, which adjusts only the degrees of freedom of regions in robot-feasible contact. Therefore, the resulting problem is sparse and block-banded and can be solved in three Gauss–Newton sweeps. Evaluated on public human motion datasets retargeted to a Unitree G1, NeCR removes penetration and cuts support-phase sliding from 20.6% to 4.4% at a cost of only 0.16 cm MPKPE, and runs at 5.9 ms per frame. Policies trained on its references track more reliably on two independently implemented trackers and transfer zero-shot to a real G1 for crawling and fall recovery. The project page is available at `https://necr-project.pages.dev/`.

1 Introduction

Leveraging large-scale human motion data [Mahmood et al. (2019); Lin et al. (2023)] to drive whole-body control of humanoid robots has become a prevailing paradigm for learning and deploying expressive humanoid behaviors [He et al. (2024; 2025b); Cheng et al. (2024)]. In this paradigm, human-to-humanoid motion retargeting maps human demonstrations into reference trajectories that

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.

the robot can execute, and the quality of that mapping bounds what downstream imitation or reinforcement learning (RL) policies can achieve [Peng et al. (2018); Araujo et al. (2025)]. Retargeting becomes particularly challenging for multi-contact motions such as crawling, kneeling, and getting up from the floor, in which the knees, shins, and forearms support the body alongside the feet. Human bodies and humanoid robots differ in their degrees of freedom (DoFs), joint limits, limb proportions, and mass distribution, so a robot may be unable to reproduce such contacts as the source human made them.

A long-standing way to bridge this embodiment gap is optimization, which models the robot's kinematics explicitly and encodes physical requirements such as joint limits and contact relationships as objective terms and constraints [Hammam et al. (2015); Ayusawa & Yoshida (2017); Yang et al. (2025)]. The resulting program, however, is high-dimensional and non-convex, so its solution is sensitive to initialization and to hand-tuned objective weights. Per-frame formulations can converge to a local optimum that matches the source in most joints while getting the supporting contact wrong [Zhao et al. (2026)], and sequence-level formulations avoid this only at a cost that grows with the sequence length [Lee et al. (2025); Zhang et al. (2026); Esteban et al. (2026)]. Both limitations stem from re-deriving the correspondence for every sequence through a hand-designed program. Learning-based methods [Aberman et al. (2020); Zhao et al. (2026); Huang et al. (2026)] instead train a network on large motion datasets to capture the cross-embodiment correspondence once, so there is no per-sequence solve to get stuck in and inference is a single forward pass. The price is that physical constraints now enter only through the training objective and are no longer enforced at inference. The resulting trajectories can exhibit foot sliding, floating, ground penetration, and physically infeasible motion [Araujo et al. (2025); Yang et al. (2025); Li et al. (2026)]. Adding contact and feasibility terms to the training loss [Huang et al. (2026)] encourages feasibility without guaranteeing it. Appending an optimization stage at inference, as in character animation [Aberman et al. (2020); Villegas et al. (2021)], does enforce it, but the contacts it enforces are binary labels read off the source human, and a robot with different proportions cannot always realize them.

Across these approaches, correction is either absent at inference, applied to the whole motion, or driven by binary contacts the robot may be unable to realize. They lack a *specification* of which DoFs the correction stage may act on, over which intervals, and with what confidence. Such a specification decides both how much of the motion is preserved and how much the correction costs.

We propose **NeCR** (**Ne**ural **C**ontact-Consistent **R**etargeting; Figure 1), which predicts this specification as a whole-body contact state and uses it to correct the output of a learned nominal retargeter. Learning thereby decides where the optimizer acts, through two components. **(a)** A temporal model predicts a multi-component *temporal contact mode* over the robot's entire surface, continuous residence probabilities for each body group describing when contact is established, maintained, and released. Predicted jointly with per-site feasibility, the mode weights each contact by whether the robot can realize it. Its continuous confidence also lets uncertain evidence produce a softer correction, where a binary label would commit to a contact that may be absent. **(b)** A structured projector uses these probabilities to decide which DoFs enter the correction and how strongly each contact is enforced. Since only a few groups are in contact at any instant, the correction stays banded and local, and three block-sparse Gauss–Newton sweeps suffice. Penetration is nearly eliminated and sliding sharply reduced, while the semantics and dynamics of the nominal motion are preserved.

Our contributions are threefold:

1. **We cast whole-body contact as a predicted specification for trajectory correction**, a continuous, multi-component temporal mode over a shared human–robot surface atlas that determines the structure of the correction program as well as the values inside it.
2. **We present NeCR, a hybrid neural–optimization framework** in which the predicted mode parameterizes a sparse, block-banded proximal projection. Correction acts only on support-relevant DoFs and its strength scales with confidence, so contact artifacts are repaired at near-zero cost in fidelity relative to the nominal trajectory it corrects.
3. **We validate NeCR on public datasets retargeted to the Unitree G1**, where it drives penetration from 58.7% to zero and cuts support-phase sliding from 20.6% to 4.4% at 0.16 cm MPKPE and 5.9 ms per frame. Under a policy-by-reference cross-validation on two independently implemented trackers, policies trained on NeCR references transfer best, and one transfers zero-shot to hardware for crawling and fall recovery.

2 Related Work

2.1 Learning-Based Humanoid Whole-Body Control

Learning-based humanoid whole-body control divides into task-driven approaches, which learn locomotion and contact-rich skills from hand-crafted objectives or command interfaces [Radosavovic et al. (2024); Gu et al. (2024); Long et al. (2025); Zhuang et al. (2025); He et al. (2025c); Xue et al. (2025); Zhang et al. (2025)], and reference-based approaches, which cast control as motion imitation [Peng et al. (2018; 2021); Luo et al. (2023)] and have been carried onto real hardware [Cheng et al. (2024); Ji et al. (2025); He et al. (2024); Fu et al. (2024); He et al. (2025b); Ze et al. (2025); He et al. (2025a)]. Recent trackers push this paradigm toward generality and scale, across heterogeneous motion distributions [Chen et al. (2025b); Yin et al. (2026)], toward downstream task control [Liao et al. (2026)], by jointly scaling data, capacity and compute [Luo et al. (2026)], and through reusable behavior representations [Zeng et al. (2025; 2026)]. Throughout, the tracked references are retargeted robot trajectories, and their fidelity bounds the behaviors a policy can acquire [Araujo et al. (2025); Li et al. (2026)].

2.2 Optimization-based Retargeting Methods

In computer graphics, retargeting is posed as inverse kinematics or spacetime optimization that preserves keypoints, end-effectors, or joint orientations under explicit kinematic constraints [Gleicher (1998); Choi & Ko (2000); Popović & Witkin (1999); Tak & Ko (2005); Lyard & Magnenat-Thalmann (2008)]. Similar formulations bridge the human–humanoid embodiment gap [Hammam et al. (2015); Ayusawa & Yoshida (2017); Penco et al. (2018); Darvish et al. (2019)]. Humanoids, however, are further constrained by joint limits, ground contacts, and self-collisions, so purely geometric matching yields floating, foot skating, ground penetration, and discontinuous joint motion [Villegas et al. (2021); Araujo et al. (2025)]. Recent work therefore strengthens the constraints imposed during optimization. GMR [Araujo et al. (2025)] uses non-uniform local scaling and staged inverse kinematics, and PHUMA [Lee et al. (2025)] adds sequence-level feasibility, grounding, and anti-skating terms. OmniRetarget [Yang et al. (2025)] and ReConForM [Cheynel et al. (2025)] preserve contact and interaction relationships across embodiments and environments. A further line moves from kinematic to dynamic feasibility, via rigid-body dynamics and multi-contact constraints [Zhang et al. (2026)], contact-implicit multiple shooting [Esteban et al. (2026)], or bilevel coupling with reinforcement learning [Müller et al. (2026)]. Contact can also be treated as a continuous variable optimized jointly with the motion instead of a fixed schedule, as in contact-invariant optimization [Mordatch et al. (2012)]. Such formulations are flexible but their objectives must be designed by hand and applied to the trajectory as a whole, and their non-convex, sequence-level structure limits both throughput at scale [Zhao et al. (2026)] and the solution any one sequence receives, motivating data-driven retargeting.

2.3 Data-driven Retargeting Methods

Data-driven methods replace iterative optimization with learned mappings across skeletons or embodiments. Character animation learns shared motion representations via latent-space alignment, skeleton-aware architectures, or cycle consistency, enabling unpaired cross-skeleton transfer [Villegas et al. (2018); Aberman et al. (2020); Lee et al. (2023); Yang et al. (2020); Hu et al. (2024); Zhang et al. (2024)], and the paradigm now extends to human-to-humanoid retargeting [Choi et al. (2020); Stanley et al. (2021); Yagi et al. (2024); Wang et al. (2024); Yan & Lee (2026)], through unpaired latent mappings [Yan et al. (2023)], guided diffusion [Cao et al. (2026)], and flow matching [Kim et al. (2025)]. The key difficulty is that matching the motion distribution does not ensure physical feasibility on the target humanoid [Yagi et al. (2024); Wang et al. (2024)]. Responses fall into two groups. Auxiliary physical losses, such as the foot-contact, foot-height, and joint-limit terms of Human2Humanoid [Huang et al. (2026)], encourage feasibility without enforcing it. Supervision from physics-refined rollouts, as in NMR [Zhao et al. (2026)] and IKMR [Chen et al. (2025a)], ties the learned mapping to a particular policy, reward, and simulator. What remains missing is an explicit temporal *specification* telling a learned retargeter where contact-driven correction is needed, while leaving the rest of the trajectory close to the transferred motion.

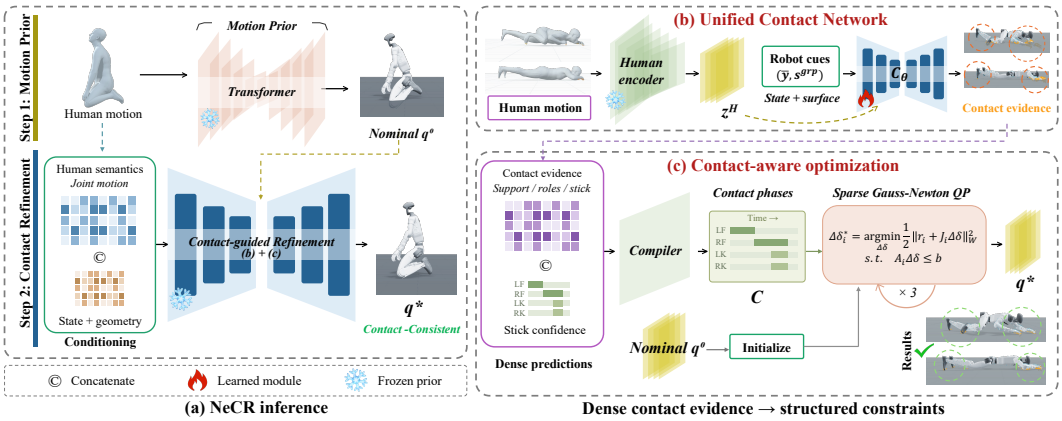

Figure 2: The NeCR inference pipeline. The nominal retargeter maps human motion to a robot trajectory. A contact network predicts temporal contact modes from the source motion, nominal state, and surface cues. These modes activate sparse contact constraints for trajectory projection.

3 Method

3.1 Problem Formulation and Method Overview

Given a human motion sequence $\mathcal{H} = \{\mathbf{h}_t\}_{t=1}^T$ of T frames sampled at 30 Hz, where $\mathbf{h}_t$ collects the root motion and root-relative body positions of frame t, our goal is a trajectory for a 29-DoF humanoid that is kinematically feasible and contact-consistent while retaining the dynamics of the source motion. We represent a robot state as $\mathbf{q}_t = (\mathbf{p}_t, \mathbf{R}_t, \boldsymbol{\theta}_t)$, where $\mathbf{p}_t \in \mathbb{R}^3$ is the root position, $\mathbf{R}_t \in \mathrm{SO}(3)$ is the root orientation, and $\boldsymbol{\theta}_t \in \mathbb{R}^{29}$ contains the joint angles, so that a trajectory is $\mathcal{Q} = \{\mathbf{q}_t\}_{t=1}^T$. A nominal retargeter can preserve the source dynamics yet still produce localized contact errors, such as brief penetration, hovering support, or unintended sliding. Correcting such errors by regularizing the full trajectory can instead distort motion that is already correct. We therefore seek a localized correction that is activated only where temporal contact evidence requires it.

NeCR decomposes this problem into nominal motion prediction, temporal contact inference, and contact-aware projection (CAP):

$$\bar{\mathcal{Q}} = f_\theta(\mathcal{H}), \qquad \mathcal{C} = g_\phi(\mathcal{H}, \bar{\mathcal{Q}}), \qquad \mathcal{Q}^* = \Pi_{\mathcal{C}}(\bar{\mathcal{Q}}), \tag{1}$$

where $\bar{\mathcal{Q}} = \{\bar{\mathbf{q}}_t\}_{t=1}^T$ is the nominal robot trajectory produced by a learned retargeter f_θ (Section 3.2); $\mathcal{C}$ is the temporal contact mode inferred by g_ϕ, which encodes for each body group g and frame t whether it is in contact, whether that contact persists, and with what confidence (Section 3.3); and $\Pi_{\mathcal{C}}$ is the correction operator these modes parameterize (Section 3.4), which projects $\bar{\mathcal{Q}}$ onto the motions consistent with $\mathcal{C}$ while remaining close to it. Throughout, $\bar{(\cdot)}$ denotes nominal quantities and $(\cdot)^*$ projected ones, and Appendix A tabulates the notation. Figure 2 illustrates the inference pipeline.

3.2 Nominal Motion Retargeting

The first stage f_θ provides a motion prior and initializes the subsequent projection. It requires a single forward pass per sequence and enforces no physical constraint. We report this nominal trajectory as *NeCR w/o CAP* in Section 4, where it is the most accurate of the trajectories we compare in pose fidelity and the worst in support-phase sliding. A transformer sequence model encodes each source frame $\mathbf{h}_t = (\mathbf{h}_t^{\mathrm{root}}, \mathbf{h}_t^{\mathrm{body}})$, with root motion and root-relative body geometry encoded separately, and decodes a robot motion feature $\bar{\mathbf{y}}_t = (\bar{\mathbf{q}}_t, \Delta\bar{\mathbf{q}}_t, \bar{\mathbf{x}}_t^{\mathrm{rel}})$ consisting of the nominal state, its per-frame displacement, and root-relative robot geometry. The separate encoding makes the features less sensitive to the human–robot size mismatch while retaining locomotion and articulated dynamics.

We train f_θ independently with a Smooth-L_1 motion loss and freeze it before training the contact model. Input features, architecture, and training details are given in Appendix B.2.

3.3 TEMPORAL CONTACT MODES

Since binary contact labels do not distinguish momentary proximity from sustained support, nor a stationary material point from a contact that may slide or roll, we instead describe contact by a continuous, multi-component temporal mode of contact residence probabilities (Appendix E.2 isolates the effect of this continuity), defined on a robot-surface atlas $\mathcal{V} = \{v_n\}_{n=1}^{N}$ of $N = 672$ nodes. Each node is rigidly attached to a link and assigned to one of $G = 12$ semantic body groups. Forward kinematics on the nominal trajectory provides signed-distance and velocity cues at the nodes, which are pooled into a group descriptor $\mathbf{s}_{t,g}^{\mathrm{grp}}$ (these cues derive from $\bar{\mathcal{Q}}$ and are implicit in Equation 1). Atlas construction is given in Appendix B.1.

For each frame and body group, the predictor outputs

$$\mathbf{c}_{t,g} = g_\phi\big(\mathcal{H}, \bar{\mathcal{Y}}, \mathbf{s}_g^{\mathrm{grp}}\big)_t = \big(c_{t,g}^{\mathrm{contact}},\, c_{t,g}^{\mathrm{support}},\, c_{t,g}^{\mathrm{stick}},\, c_{t,g}^{\mathrm{cont}}\big), \tag{2}$$

where $\bar{\mathcal{Y}} = \{\bar{\mathbf{y}}_t\}_{t=1}^{T}$ and all components are probabilities in $[0, 1]$. Together with the group modes, g_ϕ predicts $a_{t,n} \in [0, 1]$, the feasibility of node n as a contact site, and we write $\mathcal{C}$ for the collection of both. Here c^{contact} denotes contact evidence, c^{support} a sustained kinematic support phase, c^{stick} material-point residence, and c^{cont} temporal continuation. Support refers to geometric and kinematic evidence. The mode therefore specifies where contact is plausible, when it should persist, and how it may move, while $a_{t,n}$ localizes that mode on the surface. Each component drives a different part of the correction. c^{support} opens and closes the intervals on which constraints act and weights the normal residual, c^{cont} bridges brief gaps within an interval, c^{contact} and c^{stick} gate the tangential residual, and $a_{t,n}$ selects the surface patch that the residual is applied to, and with it the kinematic chain CAP is allowed to move. Architectural details are deferred to Appendix B.3.

We train g_ϕ after freezing the nominal retargeter, so that it sees the same trajectory distribution that CAP will correct. Targets are derived geometrically, with contact and its role taken from the source motion and feasibility from the nominal trajectories. Stick is additionally supervised by intended-stick labels, and g_ϕ conditions on both the source motion and the nominal trajectory. The components are trained with binary cross-entropy, and the stick component adds three terms: a boundary term upweights the frames where contact starts and ends, a counterfactual term feeds a paired input in which group-local velocities are masked, and a consistency term requires the two predictions to agree near transitions. The latter two prevent the shortcut of inferring contact from whether a body part is moving. The full objective, labels, weights, and calibration are detailed in Appendix B.3.

3.4 CONTACT-AWARE PROJECTION

The final stage is *Contact-Aware Projection* (**CAP**), the operator $\Pi_{\mathcal{C}}$ of Equation 1. It first compiles the dense modes into sparse contact events $e_k = (\tau_k^s, \tau_k^e, g_k, \mathcal{A}_k)$: the interval $[\tau_k^s, \tau_k^e)$ is obtained from $c_{t,g_k}^{\mathrm{support}}$ by hysteresis, and the connected atlas patch $\mathcal{A}_k$ is grown from the most feasible nodes, so that the two determine when and where a constraint applies. Thresholding decides only whether a constraint exists. Its strength is set by the continuous confidence, which enters the objective as a weight, so an uncertain prediction induces a softer correction rather than a hard decision.

Because the state lies on $\mathrm{SE}(3) \times \mathbb{R}^{29}$, CAP optimizes a tangent-space correction $\boldsymbol{\delta}_t$ through $\mathbf{q}_t = \bar{\mathbf{q}}_t \oplus \boldsymbol{\delta}_t$, so zero correction recovers the nominal motion exactly.

Proximal objective. CAP solves

$$\begin{aligned} \min_{\{\boldsymbol{\delta}_t\}} \quad & E_{\mathrm{prox}} + \lambda_v E_{\mathrm{vel}} + \lambda_a E_{\mathrm{acc}} + \lambda_n E_{\mathrm{contact}} + \lambda_s E_{\mathrm{stick}}, \\ \text{s.t.} \quad & \mathcal{Q} \in \mathcal{F}_{\mathrm{joint}} \cap \mathcal{F}_{\mathrm{ground}} \cap \mathcal{F}_{\mathrm{collision}}. \end{aligned} \tag{3}$$

The feasible sets $\mathcal{F}_{\mathrm{joint}}$, $\mathcal{F}_{\mathrm{ground}}$, and $\mathcal{F}_{\mathrm{collision}}$ impose joint ranges, ground clearance, and the absence of self-collisions not already present in the nominal trajectory as linearized hard constraints, independent of contact confidence. The first three terms penalize the correction itself. $E_{\mathrm{prox}} = \sum_t \|\mathbf{W}_q \boldsymbol{\delta}_t\|_2^2$ penalizes $\boldsymbol{\delta}_t$ under a diagonal inverse-tolerance weighting $\mathbf{W}_q$, and $E_{\mathrm{vel}}, E_{\mathrm{acc}}$ penalize its first and second temporal differences. Starting from the nominal trajectory, CAP keeps $\boldsymbol{\delta}$ small through E_{prox} and the trust region, and therefore refines the network's solution locally.

To make the effect of this weighting explicit, consider a single group at a single frame, ignoring the temporal terms and the coupling between groups through shared coordinates. Linearize its contact residual $\mathbf{r}$ about $\boldsymbol{\delta}_t = \mathbf{0}$, writing $\mathbf{J}$ for its Jacobian with respect to $\boldsymbol{\delta}_t$ and w for its mode weight. The stationary point of $\|\mathbf{W}_q\boldsymbol{\delta}_t\|_2^2 + \lambda_n w\|\mathbf{r} + \mathbf{J}\boldsymbol{\delta}_t\|_2^2$ is

$$\boldsymbol{\delta}_t^* = -\lambda_n w\big(\mathbf{W}_q^\top\mathbf{W}_q + \lambda_n w\,\mathbf{J}^\top\mathbf{J}\big)^{-1}\mathbf{J}^\top\mathbf{r}, \qquad \|\boldsymbol{\delta}_t^*\|_2 \le \frac{\lambda_n\|\mathbf{J}^\top\mathbf{r}\|_2}{\sigma_{\min}(\mathbf{W}_q)^2}\,w, \tag{4}$$

where the bound holds for every $w \ge 0$ because $\mathbf{J}^\top\mathbf{J} \succeq \mathbf{0}$. First, a group with $w = 0$ receives $\boldsymbol{\delta}^* = \mathbf{0}$ exactly, so regions the mode does not implicate move only through the temporal terms and the shared coordinates. Second, a group's displacement is first order in its confidence, so a false positive costs $O(w)$ rather than the $O(1)$ of a hard label. Third, the response is linear in the violation with gain $\lambda_n/\sigma_{\min}(\mathbf{W}_q)^2$, so the proximal tolerances fix in advance how much motion the projector may trade per unit of violation.

Contact objectives. Let $\mathbf{x}_{t,n}(\mathbf{q}_t) \in \mathbb{R}^3$ be the world position of atlas node n and $d(\cdot, \mathcal{G})$ the signed distance to the environment $\mathcal{G}$. For each compiled event, the normal residual pulls its active patch toward the environment surface,

$$E_{\text{contact}} = \sum_k \sum_{t=\tau_k^s}^{\tau_k^e-1} \sum_{n\in\mathcal{A}_k} w_{t,k}^n\, d(\mathbf{x}_{t,n}(\mathbf{q}_t), \mathcal{G})^2\,, \qquad w_{t,k}^n = c_{t,g_k}^{\text{support}}\,\bar{a}_{k,n}^{\gamma}, \tag{5}$$

where $\bar{a}_{k,n}$ is the node feasibility $a_{t,n}$ of Equation 2 averaged over the event interval with support-probability weights, and γ is a fixed exponent. This two-sided residual addresses both penetration and hovering, whereas the feasible set provides one-sided safety bounds. For feet, E_{stick} penalizes the frame-to-frame tangential displacement of the foot anchor node, which suppresses sliding without pinning the foot to a fixed world point. It is weighted by a foot tangential confidence $\omega_{t,f}$ built from c^{contact} and c^{stick}, and non-foot support-stick contacts are anchored per event (Appendix B.4).

Constrained update. All terms in Equation 3 are weighted squared residuals. At iteration i, CAP linearizes the residuals and feasible set and takes a constrained Gauss–Newton step,

$$\Delta\boldsymbol{\delta}_i^* = \arg\min_{\Delta\boldsymbol{\delta}} \tfrac{1}{2}\big\|\mathbf{r}_i + \mathbf{J}_i\Delta\boldsymbol{\delta}\big\|_{\mathbf{W}}^2 \quad \text{s.t.} \quad \mathbf{A}_i\Delta\boldsymbol{\delta} \le \mathbf{b}_i, \qquad \boldsymbol{\delta} \leftarrow \boldsymbol{\delta} + \Delta\boldsymbol{\delta}_i^*. \tag{6}$$

Here $\mathbf{r}_i$ and $\mathbf{J}_i$ stack the residuals of Equation 3 and their Jacobians with respect to $\boldsymbol{\delta}$ at the current iteration, $\mathbf{W}$ collects the term weights, and $\mathbf{A}_i\Delta\boldsymbol{\delta} \le \mathbf{b}_i$ is the linearization of the feasible set $\mathcal{F}_{\text{joint}} \cap \mathcal{F}_{\text{ground}} \cap \mathcal{F}_{\text{collision}}$ in Equation 3, where $\mathbf{A}_i$ stacks the constraint Jacobians and $\mathbf{b}_i$ collects the constraint margins.

The Gauss–Newton system is sparse and block-banded, since the temporal regularizers couple only neighbouring frames and the predicted modes make each contact term local to a compiled event, touching only the coordinates of the kinematic chains of the active groups. Each sweep therefore costs linearly in the sequence length with a small per-frame constant, plus a contact-dependent part proportional to the active events, and no dense coupling across the sequence arises. Sequences are solved in temporal order over overlapping windows with three Gauss–Newton sweeps each, so no frame depends on a global solve over the whole sequence. Objective weights, constraint margins, event compilation, and the inner solver are detailed in Appendix B.4 and Algorithm 1.

4 Experiments

In this section, we present a comprehensive experimental validation of **NeCR**. We benchmark the retargeted motions against state-of-the-art baselines on kinematic fidelity and motion quality in Section 4.2, and measure their impact on downstream policies in Section 4.3.

4.1 Experimental Setup

Baselines. We compare against GMR [Araujo et al. (2025)], a staged inverse-kinematics method with non-uniform scaling; OmniRetarget [Yang et al. (2025)], an optimization method that explicitly preserves interaction relationships; and NMR [Zhao et al. (2026)], a neural mapping method supervised by RL-rollout-refined data. We additionally report *NeCR w/o CAP*, the neural nominal retargeter without contact-aware projection, to isolate the effect of CAP.

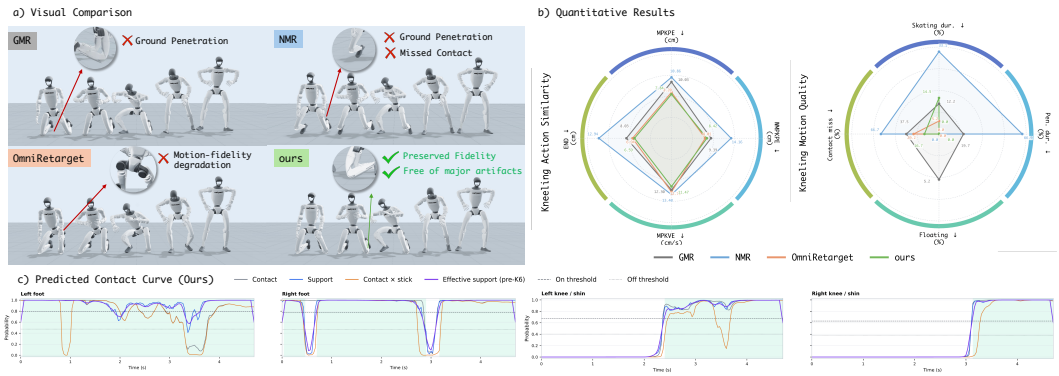

Figure 3: Kneeling-motion comparison with knee and shin contact. **(a)** Retargeted poses across methods. **(b)** Fidelity and motion-quality metrics (larger area is better). **(c)** NeCR contact modes for four support groups and CAP onset thresholds.

Downstream control framework. We evaluate the downstream impact of different retargeting methods using two independently implemented whole-body tracking policies: the BeyondMimic tracker [Liao et al. (2026)] and the Holosoma tracker [Amazon FAR (2025)] from the OmniRetarget project [Yang et al. (2025)] with proprioceptive actor observations, each with its default public configuration, the same training budget, and the same motion split (Appendix C.5).

Evaluation. We evaluate NeCR at two levels, the retargeted trajectories and the policies trained on them, on the same Unitree G1 test set. Appendix C.4 gives the metric definitions and the protocol.

Kinematic fidelity measures how closely the robot follows the human reference: **MPKPE** (mean per-keypoint position error), **W-MPKPE** (its world-frame counterpart), **MPKVE** (mean per-keypoint velocity error), **PCK@10** (keypoints within 10 cm), and **end-effector error** at the hands and feet.

Motion quality measures the contact artifacts that accurate tracking does not rule out: **foot skating** (support-phase sliding rate and peak speed), **penetration** (rate and depth over all groups in source contact), **floating** (support frames with the foot off the ground), **contact preservation** (source contact events realized on the robot), and **joint-limit violations**.

For downstream control, we report **SR** (success rate, sequences tracked to completion without falling) under policy-by-reference cross-validation, evaluating each policy on every reference distribution. Joint-error metrics and the per-duration breakdown are in Appendix D.2.

4.2 Benchmark Against Prior Retargeting Pipelines

As shown in Table 1, NeCR outperforms all baselines on both kinematic fidelity and motion quality. GMR performs staged inverse kinematics without contact reasoning, so it has the largest tracking error and penetrates on most support frames. NMR learns a direct mapping from physically refined targets; however, this does not guarantee contact consistency in its own output. OmniRetarget achieves the lowest foot skating by enforcing contact constraints on the whole trajectory, but trails NeCR on every fidelity metric. Figure 4 shows these artifacts on jump, sit, and crawl.

NeCR narrows this trade-off because the contact mode determines where the projector acts. Only groups and frames whose residence probability exceeds the onset threshold are corrected, so penetration is removed and skating is reduced to nearly the level of OmniRetarget, while the rest of the motion is unchanged. The contact mode also covers the whole body. In Figure 3(c), the knee and shin groups exceed the onset threshold together with the feet, and CAP constrains these links.

The ablation confirms that the gains come from CAP. NeCR w/o CAP has the best fidelity but the worst foot skating, and it penetrates on over half of its contact frames. CAP removes nearly all of them at a cost of only 0.16 cm MPKPE. Replacing the learned modes with height and velocity thresholds increases both skating duration and peak speed at nearly the same MPKPE (Appendix E.1).

Table 1: Retargeting quality: **(a)** kinematic fidelity and **(b)** motion quality. Values are mean ± standard deviation over sequences. First and second mark the top two methods within each panel.

(a) Kinematic fidelity

Method	MPKPE (cm)↓	W-MPKPE (cm)↓	MPKVE (cm/s)↓	End (cm)↓	PCK@10 (%)↑
GMR	9.26 ± 3.15	15.18 ± 29.55	12.55 ± 13.59	7.90 ± 4.67	67.35 ± 13.23
OmniRetarget	7.12 ± 0.91	13.47 ± 7.24	13.76 ± 9.38	5.33 ± 1.19	80.89 ± 6.52
NMR	8.46 ± 0.96	14.69 ± 10.36	13.65 ± 15.00	6.91 ± 1.99	70.81 ± 9.34
NeCR w/o CAP	**6.86 ± 0.54**	**13.01 ± 12.91**	12.03 ± 13.10	**4.26 ± 1.06**	**84.83 ± 4.61**
NeCR (Ours)	7.02 ± 0.51	13.22 ± 12.82	**11.96 ± 12.77**	4.71 ± 0.99	84.72 ± 4.49

(b) Motion quality

	Foot skating		Penetration		Floating		Contact pres.	Joint limit
Method	Dur. (%)↓	Max (cm/s)↓	Dur. (%)↓	Max (cm)↓	Dur. (%)↓	Max (cm)↓	(%)↑	(%)↓
GMR	5.11 ± 10.01	86.15 ± 90.65	69.90 ± 34.16	3.64 ± 1.13	1.09 ± 7.75	2.78 ± 7.41	91.37 ± 23.63	**0.00 ± 0.00**
OmniRetarget	**1.63 ± 3.88**	**76.65 ± 85.34**	**0.00 ± 0.00**	0.16 ± 0.08	0.15 ± 1.51	2.22 ± 5.75	91.69 ± 22.80	**0.00 ± 0.00**
NMR	15.44 ± 15.53	92.46 ± 87.91	16.45 ± 17.73	2.36 ± 1.23	**0.07 ± 0.95**	**1.91 ± 4.55**	94.01 ± 18.11	0.10 ± 0.23
NeCR w/o CAP	20.60 ± 17.20	92.82 ± 74.94	58.65 ± 24.24	2.96 ± 1.49	0.15 ± 1.21	3.53 ± 6.29	92.82 ± 18.70	0.27 ± 0.44
NeCR (Ours)	4.39 ± 5.83	77.85 ± 83.20	**0.00 ± 0.00**	**0.01 ± 0.01**	0.14 ± 1.18	3.29 ± 6.42	**94.09 ± 17.42**	**0.00 ± 0.00**

Table 2: LAFAN1 results. **(a)** NeCR against the released Unitree-LAFAN references, as mean ± standard deviation over sequences. Penetration and contact preservation are whole-body, and highlighted cells mark the better method. Full baseline comparison in Table 7. **(b)** NeCR's runtime, amortized per frame, single-threaded.

(a) Fidelity and motion quality

	Fidelity		Foot skating		Penetration		Floating		Contact pres.
Method	MPKPE (cm)↓	W-MPKPE (cm)↓	Dur. (%)↓	Max (cm/s)↓	Dur. (%)↓	Max (cm)↓	Dur. (%)↓	Max (cm)↓	(%)↑
Unitree-LAFAN	7.95 ± 0.64	**71.59 ± 38.04**	12.46 ± 7.03	**405.3 ± 174.0**	21.06 ± 11.69	4.70 ± 4.49	0.03 ± 0.10	5.42 ± 2.05	**77.47 ± 22.80**
NeCR (Ours)	**7.60 ± 0.93**	101.97 ± 47.98	**11.88 ± 6.55**	442.5 ± 184.4	**0.00 ± 0.00**	**0.08 ± 0.04**	**0.01 ± 0.09**	**3.49 ± 1.83**	76.48 ± 29.93

(b) Runtime

	Feature extraction	Nominal f_θ	Contact modes g_ϕ + compilation	CAP (3 sweeps)	Total
ms/frame	0.031	0.104	1.204	4.515	**5.855**
fps	31,798	9,592	830	222	**171**

We further compare with the publicly released Unitree-LAFAN robot references [Unitree Robotics and Contributors (2025)], a widely used locomotion baseline. On the same sequences, NeCR removes the whole-body penetration present in about one fifth of their contact samples, and achieves lower skating duration and floating with comparable contact preservation (Table 2).

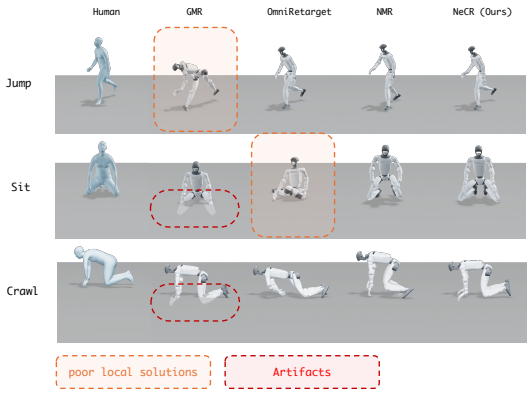

Figure 4: Jump, sit, and crawl comparisons.

Inference speed. Table 2(b) reports the runtime of NeCR on the LAFAN1 sequences. The full pipeline runs at 5.9 ms per frame single-threaded, of which the three Gauss–Newton sweeps of CAP take 4.5 ms. Sequences are processed independently, so throughput scales with the number of parallel workers. The networks are trained once and applied to LAFAN1 without retraining or fine-tuning (Appendix D.4).

4.3 Downstream Policy Training and Tracking

To verify the impact of reference trajectory quality on downstream control, we train the two trackers of Section 4.1 from scratch on the references produced by each retargeting method, holding the policy architecture, reward, and training budget fixed so that the reference trajectories are the only variable. As shown in Table 3, NeCR-trained policies achieve the highest all-reference and cross-reference average success rates on both trackers. The gains persist when the evaluation reference distribution differs from the training distribution, indicating that the improvement holds across controller-reference pairs.

Table 3: Downstream policy-by-reference cross-validation. Rows denote the reference source used to train the policy and columns denote the reference source used for evaluation. All-ref avg. averages the four evaluation references, while Cross-ref avg. excludes the matched train/evaluation reference. First and second rank within each panel and column.

(a) BeyondMimic tracker

Policy trained on	Evaluation reference				All-ref	Cross-ref
	GMR	NMR	OmniRetarget	NeCR	avg. SR	avg. SR
GMR	364/663 (54.90%)	482/663 (72.70%)	312/663 (47.06%)	444/663 (66.97%)	60.41%	62.24%
NMR	422/663 (63.65%)	585/663 (88.24%)	185/663 (27.90%)	443/663 (66.82%)	61.65%	52.79%
OmniRetarget	338/663 (50.98%)	547/663 (82.50%)	**569/663 (85.82%)**	491/663 (74.06%)	73.34%	69.18%
NeCR	**506/663 (76.32%)**	**622/663 (93.82%)**	538/663 (81.15%)	**640/663 (96.53%)**	**86.96%**	**83.76%**

(b) Holosoma tracker

Policy trained on	Evaluation reference				All-ref	Cross-ref
	GMR	NMR	OmniRetarget	NeCR	avg. SR	avg. SR
GMR	296/663 (44.65%)	385/663 (58.07%)	208/663 (31.37%)	337/663 (50.83%)	46.23%	46.76%
NMR	381/663 (57.47%)	551/663 (83.11%)	121/663 (18.25%)	396/663 (59.73%)	54.64%	45.15%
OmniRetarget	110/663 (16.59%)	334/663 (50.38%)	450/663 (67.87%)	255/663 (38.46%)	43.33%	35.14%
NeCR	**502/663 (75.72%)**	**610/663 (92.01%)**	**459/663 (69.23%)**	**612/663 (92.31%)**	**82.32%**	**78.99%**

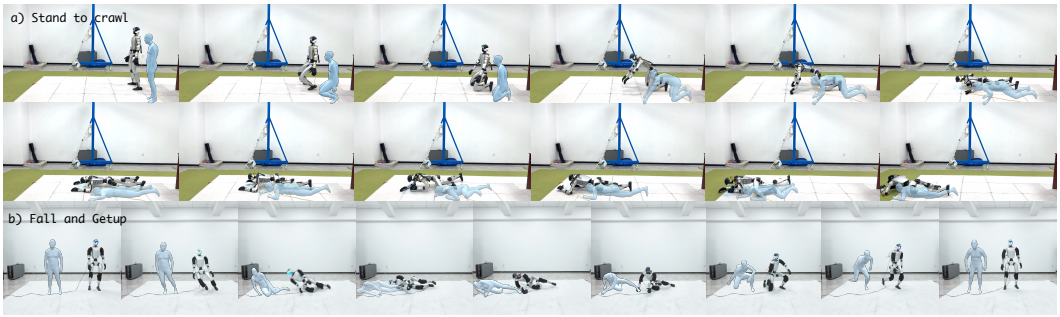

Figure 5: Sim-to-real rollouts on the Unitree G1, each read left to right and top to bottom. The blue figure is the source human motion and the robot tracks the NeCR reference retargeted from it. **(a)** Stand to crawl: from standing through kneeling into a prone crawl, loading knees, shins, and forearms in turn. **(b)** Fall and get-up: the robot goes down to the floor and pushes back to standing, handing support from torso and forearms to knees and finally to the feet.

Sim-to-real. We deploy the Holosoma tracker trained on NeCR references on a Unitree G1 with no fine-tuning on hardware. The policy tracked every crawling sequence we deployed to the end of the reference, covering the full distance of the source motion without stalling, and it also carried out a fall-and-get-up motion (Figure 5). Both pass through support sets that a foot contact schedule cannot represent, so the reference must place these contacts correctly.

5 Conclusion

We presented NeCR, a hybrid neural–optimization framework for retargeting multi-contact human motion to humanoids. A learned contact model predicts continuous whole-body contact probabilities and their feasibility for the robot, and thereby specifies where, when, and how strongly the optimizer corrects a learned nominal trajectory. The correction is sparse and block-banded and is solved in a few Gauss–Newton sweeps. It removes penetration and sharply reduces sliding at a small cost in fidelity, and policies trained on its references transfer zero-shot to a real humanoid.

Limitations. All results are on one robot, and another humanoid would need its own surface atlas and retrained networks, though the architecture carries over, and a single model spanning embodiments is left for future work. Quality degrades on highly dynamic floor motions such as kipping up, which are scarce in our training data. CAP ensures kinematic feasibility and contact consistency but leaves contact forces, friction cones, and centroidal dynamics to the tracking policy. Finally, contact is resolved against a known static surface, so movable objects and manipulation remain future work.

AI USE STATEMENT

In this work, we used generative AI tools to polish the writing for readability, to assist with editing code, and to help identify related work during the literature search. We have not used generative AI tools to formulate the research ideas, design the method or experiments, generate or process data, or interpret the results. All AI-assisted text and code were reviewed and verified by the authors, and every cited work was checked against its original source. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

The human motion data used in this work come from publicly released motion-capture datasets, including AMASS [Mahmood et al. (2019)] and LAFAN1 [Unitree Robotics and Contributors (2025)], and the motions used to train the nominal retargeter and the contact predictor come from the public BONES-SEED release [Bones Studio (2026)], all used under their respective licenses. We collected no new data from human subjects, and the data contain body motion only, with no personally identifying information beyond what the original releases provide. Our hardware experiments run retargeted motions on a humanoid robot. As with any controller executed on physical hardware, generated references should be validated in simulation before real-robot deployment. We are not aware of further ethical concerns raised by this work.

REPRODUCIBILITY STATEMENT

The method is described in Section 3 and the experimental setup in Section 4.1. The appendix provides the details needed to reproduce our results. The robot surface atlas, nominal retargeter, contact mode predictor, and Contact-Aware Projection (CAP) are specified in Appendix B, with CAP hyperparameters in Appendix B.4. Implementation details, dataset splits, baseline configurations, and evaluation metrics are given in Appendices C.1, C.2, C.3, and C.4, and the downstream control setup in Appendix C.5. The evaluation split is deterministic by subject with a fixed seed (Appendix C.2). Code, retargeted datasets, and trained policies will be released.

REFERENCES

Kfir Aberman, Peizhuo Li, Dani Lischinski, Olga Sorkine-Hornung, Daniel Cohen-Or, and Baoquan Chen. Skeleton-aware networks for deep motion retargeting. *ACM Transactions on Graphics*, 39 (4):62:1–62:14, 2020. doi: 10.1145/3386569.3392462.

Amazon FAR. Holosoma. `https://github.com/amazon-far/holosoma`, 2025. GitHub repository.

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

Ko Ayusawa and Eiichi Yoshida. Motion retargeting for humanoid robots based on simultaneous morphing parameter identification and motion optimization. *IEEE Transactions on Robotics*, 33 (6):1343–1357, 2017. doi: 10.1109/TRO.2017.2752711.

Bones Studio. BONES-SEED: Skeletal everyday embodiment dataset. `https://huggingface.co/datasets/bones-studio/seed`, 2026. Hugging Face dataset.

Sofien Bouaziz, Sebastian Martin, Tiantian Liu, Ladislav Kavan, and Mark Pauly. Projective dynamics: Fusing constraint projections for fast simulation. *ACM Transactions on Graphics*, 33(4): 1–11, 2014.

Zhefeng Cao, Ben Liu, Shunpeng Yang, Sen Li, Wei Zhang, and Hua Chen. Multi-embodiment robotic retargeting via guided diffusion model, 2026. URL `https://arxiv.org/abs/2505.20857`.

Xingyu Chen, Hanyu Wu, Sikai Wu, Mingliang Zhou, Diyun Xiang, Haodong Zhang, Yangchen Zhou, Yukang Gao, Yi Gu, and Renjing Xu. A scalable whole-body motion transfer via implicit

kinodynamic motion retargeting. *arXiv preprint arXiv:2509.15443*, 2025a. URL https://arxiv.org/abs/2509.15443.

Zixuan Chen, Mazeyu Ji, Xuxin Cheng, Xuanbin Peng, Xue Bin Peng, and Xiaolong Wang. GMT: General motion tracking for humanoid whole-body control, 2025b. URL https://arxiv.org/abs/2506.14770.

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*, 2024. doi: 10.15607/RSS.2024.XX.107.

Théo Cheynel, Thomas Rossi, Baptiste Bellot-Gurlet, Damien Rohmer, and Marie-Paule Cani. ReConForM: Real-time contact-aware motion retargeting for more diverse character morphologies. *Computer Graphics Forum*, 44(2):e70028, 2025. doi: 10.1111/cgf.70028.

Kwang-Jin Choi and Hyeong-Seok Ko. On-line motion retargetting. *The Journal of Visualization and Computer Animation*, 11(5):223–235, 2000.

Sungjoon Choi, Matt Pan, and Joohyung Kim. Nonparametric motion retargeting for humanoid robots on shared latent space. In *Proceedings of Robotics: Science and Systems*, 2020. doi: 10.15607/RSS.2020.XVI.071.

Kourosh Darvish, Yeshasvi Tirupachuri, Giulio Romualdi, Lorenzo Rapetti, Diego Ferigo, Francisco Javier Andrade Chavez, and Daniele Pucci. Whole-body geometric retargeting for humanoid robots. In *2019 IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids)*, pp. 679–686. IEEE, 2019. doi: 10.1109/Humanoids43949.2019.9035059.

Sergio A. Esteban, Jason H. K. Siu, Derrick Mach, Junheng Li, Vince Kurtz, Joel W. Burdick, and Aaron D. Ames. Shooting for contact: Contact-implicit multiple shooting for dynamic motion retargeting. *arXiv preprint arXiv:2608.03116*, 2026.

Zipeng Fu, Qingqing Zhao, Qi Wu, Gordon Wetzstein, and Chelsea Finn. HumanPlus: Humanoid shadowing and imitation from humans. In *Conference on Robot Learning (CoRL)*, 2024.

Michael Gleicher. Retargetting motion to new characters. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 33–42. Association for Computing Machinery, 1998. doi: 10.1145/280814.280820.

Xinyang Gu, Yen-Jen Wang, and Jianyu Chen. Humanoid-gym: Reinforcement learning for humanoid robot with zero-shot sim2real transfer. *arXiv preprint arXiv:2404.05695*, 2024.

Ghassan Bin Hammam, Patrick M. Wensing, Behzad Dariush, and David E. Orin. Kinodynamically consistent motion retargeting for humanoids. *International Journal of Humanoid Robotics*, 12(4): 1550017, 2015. doi: 10.1142/S0219843615500176.

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, 2024. doi: 10.1109/IROS58592.2024.10801984.

Tairan He, Jiawei Gao, Wenli Xiao, Yuanhang Zhang, Zi Wang, Jiashun Wang, Zhengyi Luo, Guanqi He, Nikhil Sobanbabu, Chaoyi Pan, Zeji Yi, Guannan Qu, Kris Kitani, Jessica K. Hodgins, Linxi Fan, Yuke Zhu, Changliu Liu, and Guanya Shi. ASAP: Aligning simulation and real-world physics for learning agile humanoid whole-body skills. In *Proceedings of Robotics: Science and Systems*, 2025a. doi: 10.15607/RSS.2025.XXI.066.

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*, pp. 1516–1540. PMLR, 2025b. URL https://proceedings.mlr.press/v270/he25b.html.

Xialin He, Runpei Dong, Zixuan Chen, and Saurabh Gupta. Learning getting-up policies for real-world humanoid robots. In *Proceedings of Robotics: Science and Systems*, 2025c. doi: 10.15607/RSS.2025.XXI.063.

Lei Hu, Zihao Zhang, Chongyang Zhong, Biao Jiang, and Shihong Xia. Pose-aware attention network for flexible motion retargeting by body part. *IEEE Transactions on Visualization and Computer Graphics*, 30(8):4792–4808, 2024. doi: 10.1109/TVCG.2023.3277918.

Tianchen Huang, Feiyang Yuan, Junchi Gu, Shurui Fang, Xiaohu Zhang, Yu Wang, Wei Gao, and Shiwu Zhang. Human2humanoid: Physics-aware cross-morphology motion retargeting for humanoid robots, 2026. URL https://arxiv.org/abs/2606.03476.

Mazeyu Ji, Xuanbin Peng, Fangchen Liu, Jialong Li, Ge Yang, Xuxin Cheng, and Xiaolong Wang. ExBody2: Advanced expressive humanoid whole-body control. In *RSS 2025 Workshop on Whole-body Control and Bimanual Manipulation: Applications in Humanoids and Beyond*, 2025. URL https://openreview.net/forum?id=rzG7xQPSCE.

Wontaek Kim, Tianyu Li, and Sehoon Ha. MoReFlow: Motion retargeting learning through unsupervised flow matching. *arXiv preprint arXiv:2509.25600*, 2025.

Kyungmin Lee, Sibeen Kim, Youngdo Lee, Minho Park, Hyunseung Kim, Dongyoon Hwang, Donghu Kim, Hojoon Lee, and Jaegul Choo. PHUMA: Physically-grounded humanoid locomotion dataset. *arXiv preprint arXiv:2510.26236*, 2025. URL https://arxiv.org/abs/2510.26236.

Sunmin Lee, Taeho Kang, Jungnam Park, Jehee Lee, and Jungdam Won. Same: Skeleton-agnostic motion embedding for character animation. In *SIGGRAPH Asia 2023 Conference Papers*, pp. 1–11. Association for Computing Machinery, 2023. doi: 10.1145/3610548.3618206.

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.

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. *Science Robotics*, 11(117):eadx8924, 2026. doi: 10.1126/scirobotics.adx8924.

Jing Lin, Ailing Zeng, Shunlin Lu, Yuanhao Cai, Ruimao Zhang, Haoqian Wang, and Lei Zhang. Motion-x: A large-scale 3d expressive whole-body human motion dataset. In *Advances in Neural Information Processing Systems*, volume 36, pp. 25268–25280, 2023. doi: 10.52202/075280-1099.

Junfeng Long, Junli Ren, Moji Shi, Zirui Wang, Tao Huang, Ping Luo, and Jiangmiao Pang. Learning humanoid locomotion with perceptive internal model. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 9997–10003, 2025. doi: 10.1109/ICRA55743.2025.11128333.

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)*, pp. 10895–10904, 2023. doi: 10.1109/ICCV51070.2023.01000.

Zhengyi Luo, Ye Yuan, Tingwu Wang, Chenran Li, Fernando Castañeda, Sirui Chen, Zi-Ang Cao, Jiefeng Li, David Minor, Qingwei Ben, Jinhyung Park, David Sami, Zi Wang, Xingye Da, Runyu Ding, Cyrus Hogg, Lina Song, Edy Lim, Eugene Jeong, Tairan He, Haoru Xue, Wenli Xiao, Simon Yuen, Jan Kautz, Yan Chang, Umar Iqbal, Linxi "Jim" Fan, and Yuke Zhu. SONIC: Supersizing motion tracking for natural humanoid whole-body control. *Science Robotics*, 11 (117):eaed4592, 2026. doi: 10.1126/scirobotics.aed4592.

Etienne Lyard and Nadia Magnenat-Thalmann. Motion adaptation based on character shape. *Computer Animation and Virtual Worlds*, 19(3–4):189–198, 2008. doi: 10.1002/cav.233.

Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *The IEEE International Conference on Computer Vision (ICCV)*, 2019. URL https://amass.is.tue.mpg.de.

Igor Mordatch, Emanuel Todorov, and Zoran Popović. Discovery of complex behaviors through contact-invariant optimization. *ACM Transactions on Graphics*, 31(4):43:1–43:8, 2012. doi: 10.1145/2185520.2185539.

David Müller, Agon Serifi, Sammy Christen, Ruben Grandia, Espen Knoop, and Moritz Bächer. ReActor: Reinforcement learning for physics-aware motion retargeting. *ACM Transactions on Graphics*, 45(4):1–11, 2026. doi: 10.1145/3811378.

Matthew Overby, George E. Brown, Jie Li, and Rahul Narain. ADMM $\supseteq$ projective dynamics: Fast simulation of hyperelastic models with dynamic constraints. *IEEE Transactions on Visualization and Computer Graphics*, 23(10):2222–2234, 2017. doi: 10.1109/TVCG.2017.2730875.

Luigi Penco, Brice Clement, Valerio Modugno, Enrico Mingo Hoffman, Gabriele Nava, Daniele Pucci, Nikos G. Tsagarakis, Jean-Baptiste Mouret, and Serena Ivaldi. Robust real-time whole-body motion retargeting from human to humanoid. In *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*, pp. 425–432. IEEE, 2018. doi: 10.1109/HUMANOIDS.2018.8624943.

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):143:1–143:14, 2018. doi: 10.1145/3197517.3201311.

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):144:1–144:20, 2021. doi: 10.1145/3450626.3459670.

Zoran Popović and Andrew Witkin. Physically based motion transformation. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 11–20. ACM Press/Addison-Wesley Publishing Co., 1999. doi: 10.1145/311535.311536.

Ilija Radosavovic, Tete Xiao, Bike Zhang, Trevor Darrell, Jitendra Malik, and Koushil Sreenath. Real-world humanoid locomotion with reinforcement learning. *Science Robotics*, 9(89):eadi9579, 2024. doi: 10.1126/scirobotics.adi9579.

Matthew Stanley, Lingfeng Tao, and Xiaoli Zhang. Robust motion mapping between human and humanoids using cycleautoencoder. In *2021 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 93–98. IEEE, 2021. doi: 10.1109/ROBIO54168.2021.9739345.

Seyoon Tak and Hyeong Seok Ko. A physically-based motion retargeting filter. *ACM Transactions on Graphics*, 24(1):98–117, 2005. doi: 10.1145/1037957.1037963.

Unitree Robotics and Contributors. Unitree LAFAN1 Retargeting Dataset. `https://huggingface.co/datasets/lvhaidong/LAFAN1_Retargeting_Dataset`, 2025. Hugging Face dataset.

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*, pp. 8639–8648, 2018.

Ruben Villegas, Duygu Ceylan, Aaron Hertzmann, Jimei Yang, and Jun Saito. Contact-aware retargeting of skinned motion. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9700–9709, 2021. doi: 10.1109/ICCV48922.2021.00958.

Tianyu Wang, Dwait Bhatt, Xiaolong Wang, and Nikolay Atanasov. Cross-embodiment robot manipulation skill transfer using latent space alignment, 2024. URL `https://arxiv.org/abs/2406.01968`.

Yufei Xue, Wentao Dong, Minghuan Liu, Weinan Zhang, and Jiangmiao Pang. A unified and general humanoid whole-body controller for fine-grained locomotion. In *Proceedings of Robotics: Science and Systems*, 2025. doi: 10.15607/RSS.2025.XXI.067.

Satoshi Yagi, Mitsunori Tada, Eiji Uchibe, Suguru Kanoga, Takamitsu Matsubara, and Jun Morimoto. Unsupervised neural motion retargeting for humanoid teleoperation, 2024. URL `https://arxiv.org/abs/2406.00727`.

Yashuai Yan and Dongheui Lee. Learning a unified latent space for cross-embodiment robot control, 2026. URL https://arxiv.org/abs/2601.15419.

Yashuai Yan, Esteve Valls Mascaro, and Dongheui Lee. ImitationNet: Unsupervised human-to-robot motion retargeting via shared latent space. In *2023 IEEE-RAS 22nd International Conference on Humanoid Robots (Humanoids)*, pp. 1–8, 2023. doi: 10.1109/Humanoids57100.2023.10375150.

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. URL https://arxiv.org/abs/2509.26633.

Zhuoqian Yang, Wentao Zhu, Wayne Wu, Chen Qian, Qiang Zhou, Bolei Zhou, and Chen Change Loy. Transmomo: Invariance-driven unsupervised video motion retargeting. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5305–5314, 2020. doi: 10.1109/CVPR42600.2020.00535.

Kangning Yin, Weishuai Zeng, Ke Fan, Minyue Dai, Zirui Wang, Qiang Zhang, Zheng Tian, Jingbo Wang, Jiangmiao Pang, and Weinan Zhang. UniTracker: Learning universal whole-body motion tracker for humanoid robots. *IEEE Robotics and Automation Letters*, 11(7):8124–8131, 2026. doi: 10.1109/LRA.2026.3692091.

Kevin Zakka. Mink: Python inverse kinematics based on MuJoCo. https://github.com/kevinzakka/mink, 2024. GitHub repository.

Yanjie Ze, Zixuan Chen, Joao Pedro Araujo, Zi ang Cao, Xue Bin Peng, Jiajun Wu, and C. Karen Liu. TWIST: Teleoperated whole-body imitation system. In *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 2143–2154. PMLR, 2025. URL https://proceedings.mlr.press/v305/ze25a.html.

Weishuai Zeng, Shunlin Lu, Kangning Yin, Xiaojie Niu, Minyue Dai, Jingbo Wang, and Jiangmiao Pang. Behavior foundation model for humanoid robots, 2025. URL https://arxiv.org/abs/2509.13780.

Weishuai Zeng, Kangning Yin, Xiaojie Niu, Shunlin Lu, Weixiang Zhong, Jiahe Chen, Feiyu Jia, Xiao Chen, Zirui Wang, Furui Xu, Ming Zhou, Kailin Li, Weinan Zhang, He Wang, Li Yi, Dahua Lin, Jiangmiao Pang, and Jingbo Wang. Scaling behavior foundation model for humanoid robots, 2026. URL https://arxiv.org/abs/2607.15163.

Chong Zhang, Wenli Xiao, Tairan He, and Guanya Shi. Wococo: Learning whole-body humanoid control with sequential contacts. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 455–472. PMLR, 2025. URL https://proceedings.mlr.press/v270/zhang25a.html.

Haodong Zhang, Zhike Chen, Haocheng Xu, Lei Hao, Xiaofei Wu, Songcen Xu, Zhensong Zhang, Yue Wang, and Rong Xiong. Semantics-aware motion retargeting with vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2155–2164, 2024. doi: 10.1109/CVPR52733.2024.00210.

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

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

Ziwen Zhuang, Shenzhe Yao, and Hang Zhao. Humanoid parkour learning. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 1975–1991. PMLR, 2025. URL https://proceedings.mlr.press/v270/zhuang25a.html.

APPENDIX TABLE OF CONTENTS

A Notation **16**

B Method Details **16**

B.1 Robot Surface Atlas 16

B.2 Nominal Retargeter 18

B.3 Contact Mode Prediction 18

B.4 Contact-Aware Projection 20

C Experimental Setup **22**

C.1 Implementation Details 22

C.2 Datasets and Splits 22

C.3 Baseline Configurations 23

C.4 Evaluation Metrics 24

C.5 Downstream Control Framework 26

D Additional Results **28**

D.1 Full LAFAN1 Comparison 28

D.2 Full Downstream Tracking Results 28

D.3 Qualitative Results 29

D.4 Inference Speed 30

E Ablation Studies **31**

E.1 Contact Estimator 31

E.2 Effect of Binarizing Contact Weights 32

E.3 Objective Terms 32

E.4 Number of Gauss–Newton Sweeps 33

A NOTATION

Table 4 collects the symbols used in Section 3.4, its preceding subsections, and the method details of Appendix B.

Table 4: Symbols used in the Method section and Appendix B.

Symbol	Meaning
T; t	sequence length in frames (30 Hz); frame index
$\mathcal{H} = \{\mathbf{h}_t\}$	human motion sequence; $\mathbf{h}_t \in \mathbb{R}^{140}$ per-frame feature
$\mathbf{q}_t = (\mathbf{p}_t, \mathbf{R}_t, \boldsymbol{\theta}_t)$	robot state: root position, root orientation, 29 joint angles
$\mathcal{Q}$; $\bar{\mathcal{Q}}$; $\mathcal{Q}^*$	trajectory; nominal trajectory from f_θ; projected output
$\bar{\mathbf{y}}_t \in \mathbb{R}^{217}$; $\bar{\mathcal{Y}}$	nominal motion feature $(\bar{\mathbf{q}}_t, \Delta\bar{\mathbf{q}}_t, \bar{\mathbf{x}}_t^{\mathrm{rel}})$; its sequence
f_θ; g_ϕ; $\Pi_{\mathcal{C}}$	nominal retargeter; contact predictor; contact-aware projector
$\mathcal{C} = (\{\mathbf{c}_{t,g}\}, \{a_{t,n}\})$	temporal contact mode: group modes and node feasibility
$\boldsymbol{\delta}_t = (\boldsymbol{\delta}_t^p, \boldsymbol{\delta}_t^R, \boldsymbol{\delta}_t^\theta) \in \mathbb{R}^{35}$	tangent-space correction: root translation, rotation vector, joint angles
$\oplus$	retraction $\mathbf{q}_t = \bar{\mathbf{q}}_t \oplus \boldsymbol{\delta}_t$: exponential map on the rotation, addition elsewhere
$\mathcal{V} = \{v_n\}$; $N = 672$	robot surface atlas; number of atlas nodes
g; $G = 12$	body group; number of groups
$\mathbf{s}_{t,g}^{\mathrm{grp}}$	pooled surface descriptor (signed distance and velocity cues) of group g
$c_{t,g}^{\mathrm{contact}}$	probability that group g touches the environment
$c_{t,g}^{\mathrm{support}}$	probability that group g is in a support-like contact phase
$c_{t,g}^{\mathrm{stick}}$	probability that the contact point of group g stays fixed on the surface
$c_{t,g}^{\mathrm{cont}}$	probability that an ongoing contact phase continues
$a_{t,n}$; $\bar{a}_{k,n}$	node contact feasibility; its support-weighted event average
$\eta_g^{\mathrm{on}}, \eta_g^{\mathrm{off}}$	calibrated per-group hysteresis thresholds
$e_k = (\tau_k^s, \tau_k^e, g_k, \mathcal{A}_k)$	compiled contact event over the half-open interval $[\tau_k^s, \tau_k^e)$
$\omega_{t,f}$; n_f; $r_{t,f}$	foot tangential confidence; foot anchor node; phase-edge ramp
$\mathbf{x}_{t,n}(\mathbf{q}_t)$	world position of node n
$\mathbf{r}_i$; $\mathbf{J}_i$; $\mathbf{W}$	stacked residual, Jacobian, and term weights of the i-th Gauss–Newton step
$\mathbf{A}_i\Delta\boldsymbol{\delta} \le \mathbf{b}_i$	linearized feasible set of Equation 3 at iteration i
$\mathcal{G}$; $d(\cdot, \mathcal{G})$	environment; signed distance, positive above the surface
$\mathbf{W}_q$	diagonal inverse-tolerance weighting of the proximal term
$\mathbf{P}_{t,f}^{\parallel}$	projection onto the local contact tangent plane
$w_{t,k}^n$; $\gamma = 0.5$	normal-contact confidence; its feasibility exponent
ϵ_g	ground activation height for clearance constraints
m; $L = 32$	solver window index; window length

B METHOD DETAILS

B.1 ROBOT SURFACE ATLAS

Contact is represented on a fixed set of $N = 672$ surface nodes distributed over the robot and partitioned into $G = 12$ groups. A group is a (semantic part, side) pair, so the five bilateral parts contribute one group per side and the two central parts contribute one group each. Table 5 lists the partition together with the robot coordinates each group depends on, and Figure 6 shows the resulting node layout on the robot alongside its SMPL-X counterpart. Every node of a group is rigidly attached to a single link, so its position depends only on the kinematic chain from the root to that link. CAP restricts each group's correction to a small set of coordinates around that chain (Table 5), which keeps the projection Jacobian sparse.

Node placement. Nodes are placed once, offline, from the compiled robot model, and are neither learned nor resampled per sequence. For each of the 12 parts we take the visual mesh attached to its link, draw an area-weighted sample of surface points, and reduce that candidate set to the target count by farthest-point selection, so coverage is near-uniform over the mesh regardless of triangle density. Each node stores its position and outward normal in the link frame.

Human counterpart. The same 12 groups are instantiated on SMPL-X as a fixed set of 2,256 template vertices, so human and robot contact are described in a shared vocabulary. A vertex is assigned to the group whose joint set carries the dominant share of its linear-blend-skinning weight,

Table 5: The $G = 12$ contact groups. Node and vertex counts are per side, so the five bilateral parts each contribute two groups. *Unlocked coordinates* are the robot coordinates CAP may move while the group carries an active contact event (the root pose is always unlocked), and *tangential DoFs* are the joints to which the group's tangential residual is restricted.

Semantic part	Sides	G1 nodes	SMPL-X verts.	Unlocked coordinates	Tangential DoFs
Foot	L, R	48	160	root, same-side leg	same-side leg
Knee/shin	L, R	64	176	root, both legs	same-side leg
Hand	L, R	64	240	root, waist, same-side arm	waist, same-side arm
Elbow/forearm	L, R	56	176	root, waist, same-side arm	waist, same-side arm
Shoulder/upper arm	L, R	32	120	root, waist, same-side shoulder, elbow	waist, same-side arm
Pelvis	centre	64	256	root, both legs, waist	root, both legs
Torso	centre	80	256	root, waist	—
Total	12 groups	672	2,256		

and within each group the required number of vertices is again chosen by farthest-point selection, with groups kept disjoint.

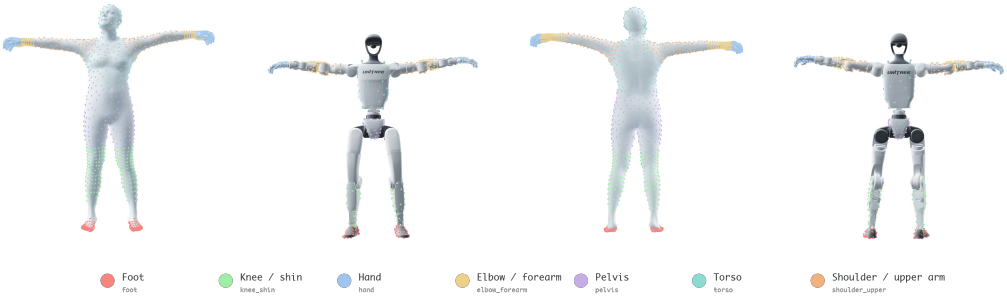

Figure 6: The contact atlas and its human counterpart, shown in the T-pose from the front and from the back. Each pair places the SMPL-X template vertices next to the $N = 672$ surface nodes on the Unitree G1, coloured by semantic part. The five bilateral parts are split into a left and a right group, so the seven colours in the legend correspond to the $G = 12$ groups of Table 5. Both surfaces are partitioned by the same seven parts, so contact evidence predicted on the source motion can be attached directly to robot geometry.

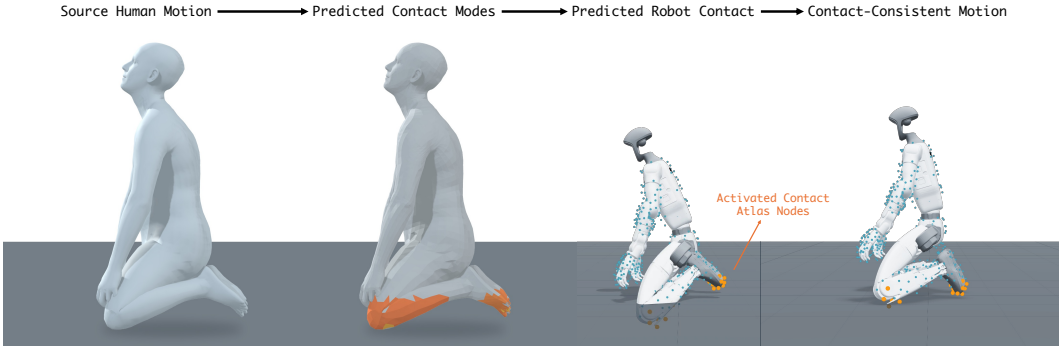

Figure 7: An activated atlas on a kneeling sequence, read left to right. Contact modes predicted from the source motion and the nominal rollout mark the supporting region in orange. On the robot, the activated nodes (orange) form a subset of the full node set (cyan), and the projection then returns the contact-consistent robot motion. Here the lower-leg and foot groups carry the support while the hands stay free. The prediction and projection stages are detailed in Appendices B.3 and B.4.

Figure 7 shows a predicted contact mode once it has been attached to the atlas, on a kneeling frame in which the lower-leg and foot groups are active and the remaining groups stay free. The projection in Section 3.4 operates directly on this activated node set.

B.2 NOMINAL RETARGETER

Architecture. The nominal retargeter is a transformer sequence model with 8 layers, 8 attention heads, width 512 and a maximum context of 1,024 frames, totalling 53.0M parameters. Source motion enters as raw SMPL/SMPL-X parameters. An attention encoder of width 512 with a temporal downsampling factor of four maps it to the per-frame human motion feature $\mathbf{h}_t$, made of the horizontal root displacement, the 6D root orientation, the root-relative body positions, and the body displacements. The transformer decodes a per-frame robot motion feature from which the output is read off, namely the root translation, the root orientation, and the 29 joint angles of the Unitree G1. Human and robot feature normalizers are frozen train-split statistics, and velocity slices follow a per-frame displacement convention.

Training. Regression targets are Unitree G1 trajectories selected from BONES-SEED [Bones Studio (2026)] to span a wide range of motion types. The objective is a Smooth-L_1 regression on the robot motion feature ($\beta = 1$, weight 1.0). Optimization uses AdamW ($\mathrm{lr} = 2 \times 10^{-4}$, $(\beta_1, \beta_2) = (0.9, 0.99)$, no weight decay) for 500 epochs with a 500-iteration linear warmup and cosine decay to 10^{-5}, batch size 128, clips of 60–300 frames cropped to a multiple of four frames. No optimizer or projection stage is in the loop at this point, so the frozen nominal rollouts form a well-defined target distribution for the contact predictor of Appendix B.3.

B.3 CONTACT MODE PREDICTION

Inputs. Each frame supplies three tensors. The source motion is the 140-dimensional human feature $\mathbf{h}_t$ of Appendix B.2, and the nominal rollout is the 217-dimensional robot feature $\bar{\mathbf{y}}_t$. Surface geometry is observed per atlas node as five channels: the geometric contact and stick likelihoods of the node under the nominal rollout, its signed height normalized by 0.20 m and clipped to $[-0.5,2]$, its vertical speed clipped to ± 2 m/s, and its tangential speed clipped to [0,2] m/s. The group descriptor $\mathbf{s}^{\mathrm{grp}}_{t,g}$ of Equation 2 is the concatenation of the per-channel mean and maximum over the nodes of group g, so it is ten-dimensional. Human and robot features are normalized with frozen train-split statistics.

Supervision. Labels come from two sides. Contact and its role are decided on the source human motion, so they describe the contact intended by the source motion, independently of what the nominal rollout realizes. The nominal rollout enters afterwards, as a feasibility audit deciding whether the robot can carry a given contact at all. No projection is run while labels are produced. Each frame and group carries a five-way role label (none, touch, support-stick, support-roll, or slide), of which support-stick and support-roll count as support. Alongside it we store per-group support and continuation targets and a per-node feasibility target over the 672 atlas nodes. Stick carries one further label. Frames whose source contact is stationary while the nominal foot moves are the *intended-stick* hard positives. They select the groups on which the counterfactual velocity masking of Equation 9 is applied, and recall on them is tracked during validation.

Human-side contact labels. For each SMPL-X atlas vertex i we take its height h_i above the estimated floor, its outward normal $\mathbf{n}_i$, and its finite-difference velocity $\mathbf{v}_i$. The contact likelihood is a product of a soft distance gate, a normal term, and a vertical-speed term,

$$p_i^{\mathrm{contact}} = \sigma\left(\frac{0.025 - |h_i|}{0.008}\right)\left[0.1 + 0.9\sqrt{\max(-n_{i,\mathrm{up}}, 0)}\right]\exp\left[-\tfrac{1}{2}\left(\frac{|v_{i,\mathrm{up}}|}{0.50}\right)^2\right], \tag{7}$$

with heights in metres and speeds in m/s. Contact fades over a 2.5 cm band with a 0.8 cm temperature, downward-facing surfaces count in full while upward and lateral ones keep a floor weight of 0.1, and fast vertical motion suppresses the label. Material residence multiplies in two further Gaussians on the vertical and tangential speed, both at a scale of 0.15 m/s,

$$p_i^{\mathrm{stick}} = p_i^{\mathrm{contact}} \exp\left[-\tfrac{1}{2}\left(\frac{|v_{i,\mathrm{up}}|}{0.15}\right)^2 - \tfrac{1}{2}\left(\frac{\|\mathbf{v}_{i,\tan}\|}{0.15}\right)^2\right]. \tag{8}$$

Both are soft, so no height threshold declares contact on its own. The floor is estimated from the frames in which the pelvis is above its 75th percentile and at least one foot moves no faster than 0.25 m/s. Over those frames we take the median of the lowest of six toe and heel landmarks, falling

back to the 10th percentile of that per-frame minimum when too few frames qualify. A geometric patch keeps the vertices with $p^{\text{contact}} \geq 0.50$, takes mesh-connected components within one body group, and discards components with fewer than three atlas anchors.

Contact roles. Roles follow the material patches over time. Two observations ℓ frames apart join the same track when their world tangential displacement is at most 0.15ℓ m and either their patches overlap with IoU ≥ 0.10 or their material-surface distance is at most 0.06 m. They are the *same* material patch when IoU ≥ 0.50 and that distance is at most 0.02 m. A contact is *support-stick* when it stays on the same material patch and moves at most 0.01ℓ m in the world tangent plane. It is *support-roll* when the material patch changes but stays within 0.06 m of the previous one and the world tangential displacement is at most 0.02ℓ m, and *slide* when the material patch is unchanged but the displacement exceeds 0.01ℓ m. Everything else, such as a newly appearing patch or one that cannot be associated, is *touch*. At 30 Hz and $\ell = 1$ the two displacement gates correspond to roughly 0.30 and 0.60 m/s, although the test is on displacement scaled by the frame gap and not on an explicit speed. Roles are smoothed by a majority filter of radius two frames, and runs shorter than three frames revert to touch.

Height enters only as a reliability tier, never as a support criterion. Writing $q_{0.05}(h)$ for the fifth percentile of the height over the contact component and its two-hop mesh neighbourhood, a segment is *normal* when $q_{0.05}(h) \geq -2.5\,\text{cm}$, *moderate* when $-5\,\text{cm} \leq q_{0.05}(h) < -2.5\,\text{cm}$, and *deep* below that. Moderate segments are down-weighted to 0.5, and for deep segments the role and robot-patch supervision are masked out entirely.

Robot-side feasibility. A human support segment becomes a robot target only if the robot can hold it. For each segment we compile at most eight atlas nodes on the same body and audit up to nine frames near its start, middle, and end by linearizing the contact system on the nominal rollout. The gates are a contact-height upper bound of 1.0 cm, a full-atlas ground floor of -0.25 cm, a ground activation height of 2.5 cm, and a stick tangential band of 1.0 cm, with residual slacks of at most 2.5 mm for contact, ground, and collision and 1.0 cm for stick. A support-stick segment must pass all four gates, and support-roll is exempt from the stick gate. A sampled frame counts as feasible only when every relevant slack passes, and the segment is feasible when at least 0.80 of its sampled frames are. That segment-level decision is also the per-node target. Every node of the compiled patch takes it, and only those nodes have their loss mask opened. Nodes outside the patch are ignored, not supervised as infeasible negatives, so the node head learns which sites are usable without being penalized for sites that were simply not selected. At inference the feasibility threshold is applied to the support-weighted mean over each candidate patch.

Training pool. Training uses a frozen pool of 3,323 BONES-SEED clips (849,920 frames at 30 Hz), split into 2,428 training, 421 validation, and 474 test clips. The pool, its nominal rollouts, and its labels are regenerated and checked against these counts before every run, so that no partially regenerated pool enters training.

Architecture. g_ϕ has three parts. The human branch is a bidirectional transformer over the source-motion feature with six layers of width 384, eight heads, feed-forward width 1536, and node embeddings of width 128. It predicts the human surface contact map, the group contact probability c^{contact}, and a human-side continuation, and it is trained first and frozen. The robot branch embeds the nominal motion feature, passes it through a two-layer temporal transformer of the same width, and combines it with the human features by a gated residual fusion whose gate is initialised near zero, so training starts from the human-only representation and learns how much nominal-specific evidence to admit. Per-node feasibility is a scaled dot product between a per-frame query and a per-node key, where the key combines the node's fixed geometry with learned group and side embeddings and a small encoder over per-node surface observations, plus a learned per-node bias. Group-level heads read the same fused representation for support, continuation, and an auxiliary five-way contact role (none, touch, support-stick, support-roll, slide). The role is not part of Equation 2. It is used at compilation time, to reject phases whose role is not a support role and to select the anchored form of the tangential term for the phases it marks support-stick. The third part owns tangential stick. It is a separate temporal decoder, a two-layer causal LSTM of hidden width 256 with 1.16M parameters. Inference runs one network call per 256-frame chunk, with overlapping chunks averaged.

Training. The three parts are trained in order, each with the earlier parts frozen. The human branch is trained on the human-side labels with a focal binary cross-entropy plus a soft Dice term on the surface contact map, the same pair at weight 0.5 on the group contact probabilities, and a focal binary cross-entropy at weight 0.25 on continuation. The robot branch is trained on the nominal-rollout labels with a focal binary cross-entropy on node feasibility, focal binary cross-entropy plus soft Dice on support, categorical cross-entropy at weight 0.5 on the auxiliary role, focal binary cross-entropy at weight 0.25 on continuation, and a group-local ranking term at weight 0.5 that orders feasible nodes above infeasible ones within each group. Both branches use AdamW at an initial learning rate of 2×10^{-4}. The human branch runs for up to 80 epochs, with the checkpoint chosen on the validation split (epoch 74). The robot branch, trained on top of it, follows a 15-epoch cosine schedule from which epoch 10 was selected. The stick decoder is trained last with

$$\mathcal{L}_{\text{contact}} = \mathcal{L}_{\text{BCE}} + \lambda_b \mathcal{L}_{\text{boundary}} + \lambda_{\text{cf}} \mathcal{L}_{\text{counterfactual}} + \lambda_c \mathcal{L}_{\text{consistency}}, \tag{9}$$

where the boundary term upweights the frames where contact starts and ends. We optimize with AdamW with weight decay 10^{-3} and cosine decay, batch size 8, in two stages of 15 epochs each. A foot-only stage runs at learning rate 5×10^{-4} decaying to 10^{-5}, and a whole-body stage warm-started from the best foot checkpoint runs at 2×10^{-4} decaying to 4×10^{-6}. All upstream tensors stay frozen throughout. The objective of Equation 9 uses $\mathcal{L}_{\text{BCE}}$ at weight 1.0 with $(\lambda_b, \lambda_{\text{cf}}, \lambda_c) = (0.25, 0.5, 0.1)$. The counterfactual term is formed from a paired view that drops group-local velocity evidence with probability 0.5, and the consistency term ties that view to the primary one within a radius of two frames around contact transitions, where the model is most likely to rely on velocity cues.

Predictor accuracy. On the validation split of this pool, at the deployment operating point, the support head reaches a pooled precision of 0.917 at a recall of 0.816 over group–frame samples, and leaves 2.3 groups active per frame on average. The stick head reaches an area under the precision–recall curve of 0.9996 and an F_1 of 0.990 at threshold 0.5 on the feet, and 0.9956 and 0.966 on the remaining groups, with an expected calibration error of 0.046. The frames that matter most for CAP are those in which the source contact is stationary but the nominal foot is not. Recall on them is 0.934, against 0.997 on the remaining positives, so the predictor keeps flagging stick where the nominal rollout has already lost it.

Calibration. After training we fit per-group temperatures on the validation split only, with no slope or bias terms, and select the deployment operating point on the same split. The resulting onset thresholds lie between 0.49 and 0.79 and the release thresholds between 0.29 and 0.47, so each group is gated on its own scale instead of a single global value.

B.4 Contact-Aware Projection

Sparse contact compilation. Compilation turns per-frame probabilities into a small set of contact phases, each carrying a concrete patch of atlas nodes. Per-group evidence is the support probability tempered by a patch-level feasibility summary (exponent 0.5) and smoothed over five frames. It is decoded into phases by hysteresis with the per-group onset and release thresholds fitted on the validation split. Phases shorter than a few frames are discarded, and foot phases separated by a brief gap are merged when the continuation probability c^{cont} over the gap is high. Within a phase, node scores are the support-weighted temporal average of per-node feasibility, and the patch grows from the highest-scoring node along a fixed surface-adjacency graph of the atlas, so patches stay contiguous on the surface. A phase is dropped if its predicted role is not a support role or if its patch feasibility falls below a fixed threshold. Foot tangential confidence $\omega_{t,f}$ is the product of contact, stick, and anchor-node feasibility, ramped at each phase edge so that stick engages and releases gradually. We do not scale this confidence by nominal tangential speed, because fast sliding is where correction is most needed and down-weighting it would suppress the strongest violations.

Tangential terms. The foot tangential term of Section 3.4 penalizes material displacement between consecutive frames,

$$E_{\text{stick}}^{\text{foot}} = \sum_{f} \sum_{t=2}^{T} \omega_{t,f} \left\| \mathbf{P}_{t,f}^{\parallel} \big(\mathbf{x}_{t,n_f}(\mathbf{q}_t) - \mathbf{x}_{t-1,n_f}(\mathbf{q}_{t-1}) \big) \right\|_2^2, \tag{10}$$

Algorithm 1 Contact-Aware Projection (CAP). Hyperparameter values (L, I, the hysteresis thresholds, and the ramp length) are given in the text of this appendix.

Require: nominal trajectory $\bar{\mathcal{Q}} = \{\bar{\mathbf{q}}_t\}_{t=1}^{T}$; contact modes $\{\mathbf{c}_{t,g}\}$ and node feasibility $\{a_{t,n}\}$; atlas $\mathcal{V}$; environment $\mathcal{G}$
Ensure: projected trajectory $\mathcal{Q}^*$
// Stage 1: sparse contact compilation
$\mathcal{E} \leftarrow$ hysteresis on c^{support} with $(\eta_g^{\text{on}}, \eta_g^{\text{off}})$, merging short gaps by c^{cont}
for all events $e_k \in \mathcal{E}$ **do**
  $\bar{a}_{k,n} \leftarrow$ support-weighted node feasibility over $[\tau_k^s, \tau_k^e)$
  $\mathcal{A}_k \leftarrow$ connected patch grown from $\arg\max_n \bar{a}_{k,n}$; drop e_k if role or feasibility invalid
end for
$\omega_{t,f} \leftarrow c_{t,f}^{\text{contact}} c_{t,f}^{\text{stick}} a_{t,n_f} r_{t,f}$ for each foot f ▷ anchor $n_f = \arg\max_{n \in \mathcal{A}_k} \bar{a}_{k,n}$; ramp $r_{t,f}$ at phase edges
// Stage 2: windowed block-sparse Gauss–Newton
$\mathcal{Q} \leftarrow \bar{\mathcal{Q}}$
for all windows m of L frames in temporal order, stride $L/2$ **do**
  for $i = 1$ to I **do**
    $(\mathbf{r}_i, \mathbf{J}_i, \mathbf{A}_i, \mathbf{b}_i) \leftarrow$ linearize $\mathbf{x}_{t,n}$ about $\mathcal{Q}$
    $\Delta\boldsymbol{\delta} \leftarrow$ local–global solve of equation 6 ▷ global step factored once per window
    $\mathcal{Q} \leftarrow \mathcal{Q} \oplus \Delta\boldsymbol{\delta}$
  end for
  $\boldsymbol{\delta}^{(m)} \leftarrow$ window correction; carry overlap frames forward as consensus
end for
$\boldsymbol{\delta} \leftarrow$ ramp-weighted blend of $\{\boldsymbol{\delta}^{(m)}\}$ over covering windows
$\mathcal{Q}^* \leftarrow \bar{\mathcal{Q}} \oplus \boldsymbol{\delta}$
return $\mathcal{Q}^*$

with $\mathbf{P}_{t,f}^{\parallel} = \mathbf{I} - \mathbf{n}_{t,f}\mathbf{n}_{t,f}^{\top}$ projecting onto the local contact tangent plane of normal $\mathbf{n}_{t,f}$, and n_f the foot anchor node. Penalizing successive displacement lets a contact translate slowly without accumulating the unbounded restoring force that a fixed world anchor would produce. Non-foot contacts use a phase-level anchor instead. For each event e_k whose predicted role is support-stick, the horizontal position of its anchor node n_k is held at $\mathbf{a}_k$, the median of that node's world position over the first three frames of the event,

$$E_{\text{stick}}^{\text{anchor}} = \sum_k \sum_{t=\tau_k^s}^{\tau_k^e - 1} c_{t,g_k}^{\text{support}} \left\| \mathbf{P}^{\parallel} \left(\mathbf{x}_{t,n_k}(\mathbf{q}_t) - \mathbf{a}_k \right) \right\|_2^2, \tag{11}$$

with $\mathbf{P}^{\parallel}$ the projection onto the ground plane and a residual scale of 2 cm. The anchor is set afresh for every event, so a contact that is released and re-established is not pulled back to its earlier location.

Unlocked coordinates. Within a window, the optimization variables are the root pose together with the union, over all groups carrying an active event in the window, of the coordinates listed in Table 5. All other joints keep their nominal values. Normal residuals use the full Jacobian over these variables, whereas tangential residuals are further restricted to the group's own limb. Few groups are active at once (2.3 per frame on average on the validation split), so the per-frame block of the Gauss–Newton system stays small and groups couple only through the coordinates they share, chiefly the root and the waist.

Objective and tolerances. The proximal tolerance scales in $\mathbf{W}_q$ are 4 cm for root translation and 0.15 rad for root rotation and joint coordinates. The objective weights in Equation 3 are $\lambda_v = 1$ and $\lambda_a = 0.5$ for the temporal terms, both with a scale of 0.08, $\lambda_n = 0.25$ for normal contact, and $\lambda_s = 0.10$ for the tangential stick terms, with residual scales of 2 cm for general contact and 3.3 mm for foot stick. The feasibility exponent in Equation 5 is $\gamma = 0.5$.

Ground and self-collision constraints. Ground clearance is imposed on the lowest atlas node of every link whose lowest node lies within $\epsilon_g = 2.5$ cm of the ground and, on links carrying an active support event, on all nodes within 5 mm of that lowest node. Each selected node is constrained to non-negative height. Self-collision is imposed as a linearized signed-distance constraint on every

geometry pair the collision model reports as touching, excluding the wrist and hand geometries and pairs already in contact in the nominal trajectory. Both families are re-linearized at every sweep.

Trust region and joint limits. Every per-frame correction is box constrained, and the box projection enforces the trust region and the joint limits exactly at every update. Ground clearance and self-collision enter through their per-sweep linearization, so their residual violation is bounded by the linearization error (cf. the maximum penetration depth in Table 1). Each bound is the tighter of a trust region and the distance remaining to the robot's own joint limit, where the trust region is 8 cm for root translation, 10° for root rotation, 0.20 rad for leg joints, 0.12 rad for arm joints, and 0.08 rad for waist joints. Limits are therefore enforced inside the solver, and the joint-limit violation rate in Appendix C.4 is measured against a tolerance of 10^{-5} rad to verify the constraint.

Windowing and inner solve. Sequences are solved in 32-frame windows with a 16-frame overlap and three Gauss–Newton sweeps per window. Each sweep's constrained subproblem Equation 6 is solved by a local–global splitting in the manner of projective dynamics [Bouaziz et al. (2014); Overby et al. (2017)]. The global step factors the block-banded system once per window and reuses the factorization across inner iterations. The local step applies the trust region and joint limits as closed-form box projections and the linearized ground and self-collision rows as closed-form half-space projections, with scaled dual variables in the manner of ADMM [Overby et al. (2017)]. Primal and dual variables are warm-started from the preceding window. All reported results and timings use this solver. Overlaps are reconciled in temporal order. The committed solution from the earlier window supplies the reference correction over the shared frames, and the two solutions are blended with a linear ramp that is open at both ends, so no frame is decided by a window that has not yet seen it. A window whose inner solve does not converge is re-solved with a general-purpose QP solver (ProxQP), and a window that ProxQP cannot solve either is reported as infeasible.

C Experimental Setup

C.1 Implementation Details

NeCR is implemented in PyTorch and MuJoCo. Contact-aware projection uses the sparse Gauss–Newton scheme of Appendix B.4, with the projective-dynamics inner solver described there. All neural networks are trained on a single NVIDIA GeForce RTX 5080 GPU. All downstream reinforcement learning policies are trained on NVIDIA GeForce RTX 4090 GPUs with the same training configuration and computational budget.

C.2 Datasets and Splits

Evaluation set. We use 7,319 AMASS [Mahmood et al. (2019)] sequences at 30 Hz, split deterministically by subject. Within each AMASS subset, subjects are sorted, shuffled with seed 3407, and assigned to validation and test at roughly 10% each, with the remainder for training. This gives 6,000 training (2,407,260 frames, 258 subjects), 656 validation (339,742 frames, 35 subjects), and 663 test sequences (239,888 frames, 35 subjects). The split is subject-disjoint, so no subject and no source sequence is shared between training and test or between validation and test, and no sequence was selected or removed by hand. Every compared method returns an output for every test sequence, so none is dropped from the comparison. The test sequences come from ACCAD (17), BMLmovi (170), CMU (185), DFaust (12), DanceDB (2), EKUT (22), Eyes_Japan (15), GRAB (145), HDM05 (29), HUMAN4D (13), HumanEva (10), KIT (33), MoSh (6), SFU (2), and TotalCapture (2). By duration, 195 are shorter than 5 s, 298 are between 5 and 15 s, and 170 are 15 s or longer. AMASS serves only evaluation and downstream control. Its test split is the test set on which every retargeting result is reported, and its training and validation splits are used only to train and validate the downstream tracking policies of Appendix C.5.

Training data. Both networks are trained on BONES-SEED [Bones Studio (2026)], the nominal retargeter as described in Appendix B.2 and the contact predictor on the pool of Appendix B.3. Neither sees AMASS, so no test sequence contributes to their training, contact labels, or calibration.

C.3 Baseline Configurations

The three baselines cover the two families of retargeting methods. GMR and OmniRetarget solve an optimization over the robot configuration at every frame, whereas NMR predicts the whole robot trajectory with a network. Table 6 summarizes how they differ, and the paragraphs below state each formulation. Throughout, $\mathbf{q}_t$ is the robot configuration, $f_i^p(\mathbf{q}_t)$ and $f_i^R(\mathbf{q}_t)$ are the forward kinematics of the i-th robot keypoint's position and orientation, and $\mathbf{p}_{t,i}^{\mathrm{src}}$ and $\mathbf{R}_{t,i}^{\mathrm{src}}$ are the position and orientation of the corresponding human keypoint after each method's own scaling. We run GMR and OmniRetarget ourselves with their released code and default Unitree G1 configurations, without changing any setting, and use the released pre-trained model for NMR. All methods retarget to the same 29-DoF Unitree G1 and are scored on the same test set at 30 Hz with the evaluators of Appendix C.4.

Table 6: Baseline retargeting formulations compared with NeCR.

Method	Optimization type	Primary objective	Hard constraints	Contact handling
GMR	Per-frame differential IK	Keypoint position & orientation matching	Joint limits	Global height offset
OmniRetarget	Per-frame sequential SOCP	Interaction-mesh deformation	Joints, velocity, collision	Stance foot pinned
NMR	Single network pass	Learned from RL-refined references	None at inference	Implicit in training data
NeCR	Network pass + sparse projection	Proximity to nominal trajectory	Joints, ground, collision	Predicted temporal contact modes

GMR. GMR [Araujo et al. (2025)] performs retargeting by solving an inverse kinematics problem at each frame. It first scales the human keypoints non-uniformly, with a per-body factor s_i on the root-relative position and a single factor s_{root} on the root translation,

$$\mathbf{p}_{t,i}^{\mathrm{src}} = \frac{h}{h_{\mathrm{ref}}}\Big(s_i\big(\mathbf{p}_{t,i}^{\mathrm{human}} - \mathbf{p}_{t,\mathrm{root}}^{\mathrm{human}}\big) + s_{\mathrm{root}}\,\mathbf{p}_{t,\mathrm{root}}^{\mathrm{human}}\Big), \tag{12}$$

where h is the human height and h_{ref} a reference height. At each timestep, GMR then finds the robot configuration that matches the source keypoint positions and orientations via

$$\mathbf{q}_t^* = \arg\min_{\mathbf{q}_t} \sum_i w_i^R \big\|\mathbf{R}_{t,i}^{\mathrm{src}} \ominus f_i^R(\mathbf{q}_t)\big\|_2^2 + \sum_i w_i^p \big\|f_i^p(\mathbf{q}_t) - \mathbf{p}_{t,i}^{\mathrm{src}}\big\|_2^2 \quad \text{s.t.} \quad \mathbf{q}_{\min} \le \mathbf{q}_t \le \mathbf{q}_{\max}. \tag{13}$$

The program is solved in two stages with the differential IK library Mink [Zakka (2024)]. The first stage uses all orientations but only the hand and foot positions, and the second, initialized from the first, uses the positions of all keypoints under a different weighting. Each stage runs for at most ten iterations, and each frame is warm-started from the previous one. After the whole sequence is retargeted, the minimum height of any robot body over the sequence is subtracted from the root translation. No contact or non-penetration term appears in Equation 13, so ground contact is corrected only by this global offset.

Algorithm 2 Retarget Robot Motion (GMR)

Require: Robot model, source keypoint positions $\{\mathbf{p}_{0:T,i}^{\mathrm{src}}\}$ and orientations $\{\mathbf{R}_{0:T,i}^{\mathrm{src}}\}$
Ensure: $\mathbf{q}_{0:T}$
1: **for** $t = 0, \dots, T$ **do**
2: $\quad\mathbf{q}_t \leftarrow$ Solve IK equation 13 with hand and foot positions, warm-started from $\mathbf{q}_{t-1}$
3: $\quad\mathbf{q}_t \leftarrow$ Solve IK equation 13 with all keypoint positions, warm-started from $\mathbf{q}_t$
4: **end for**
5: Subtract $\min_{t,j} f_j^p(\mathbf{q}_t)_z$ from the root height of every frame

OmniRetarget. OmniRetarget [Yang et al. (2025)] scales the human keypoints uniformly by the robot-to-human height ratio and builds an interaction mesh by Delaunay tetrahedralization of the keypoints together with points sampled on the environment. With $L(\mathbf{p}_{t,i}) = \mathbf{p}_{t,i} - \sum_{k\in\mathcal{N}(i)} w_{ik}\,\mathbf{p}_{t,k}$ the Laplacian coordinate of vertex i under uniform weights $w_{ik} = 1/|\mathcal{N}(i)|$, it operates frame by frame by minimizing the Laplacian deformation of the mesh subject to hard kinematic constraints,

$$\begin{aligned}\mathbf{q}_t^* = \arg\min_{\mathbf{q}_t} \sum_i \big\|L(\mathbf{p}_{t,i}^{\mathrm{src}}) - L\big(f_i^p(\mathbf{q}_t)\big)\big\|_2^2 + \|\mathbf{q}_t - \mathbf{q}_{t-1}\|_{\mathbf{Q}}^2 \\ \text{s.t.}\quad \phi_k(\mathbf{q}_t) \ge 0\ \ \forall k, \qquad \mathbf{q}_{\min} \le \mathbf{q}_t \le \mathbf{q}_{\max}, \qquad \mathbf{v}_{\min}\,\Delta t \le \mathbf{q}_t - \mathbf{q}_{t-1} \le \mathbf{v}_{\max}\,\Delta t, \\ \mathbf{p}_t^F(\mathbf{q}_t) = \mathbf{p}_{t-1}^F\ \ \forall \text{ stance feet},\end{aligned} \tag{14}$$

where ϕ_k is the signed distance of collision pair k, including robot–ground pairs, $\mathbf{Q}$ weights temporal smoothness, and a foot is in stance when the horizontal velocity of the corresponding source foot is below 1 cm/s. At each frame the program is linearized around the current iteration and the increment is obtained from a second-order cone program with a trust region on the step, repeated for at most ten iterations. Non-penetration and foot sticking are therefore hard constraints, but the stance decision is a binary threshold on the source motion, and all joints and the root remain free at every frame.

Algorithm 3 Retarget Robot Motion (OmniRetarget)

Require: Robot model, source keypoint positions $\{\mathbf{p}^{\mathrm{src}}_{0:T,i}\}$, environment points
Ensure: $\mathbf{q}_{0:T}$
1: **for** $t = 0, \dots, T$ **do**
2: $\quad\bar{\mathbf{q}} \leftarrow \mathbf{q}_{t-1}$
3: $\quad$**for** $n = 1, \dots, 10$ **do**
4: $\quad\quad\Delta\mathbf{q}^* \leftarrow$ Solve SOCP of equation 14 linearized at $\bar{\mathbf{q}}$, with trust region $\|\Delta\mathbf{q}\|_2 \le \varepsilon$
5: $\quad\quad\bar{\mathbf{q}} \leftarrow \bar{\mathbf{q}} + \Delta\mathbf{q}^*$
6: $\quad$**end for**
7: $\quad\mathbf{q}_t \leftarrow \bar{\mathbf{q}}$
8: **end for**

NMR. NMR [Zhao et al. (2026)] replaces the per-frame optimization with a network F_ψ that maps the whole human sequence to the whole robot sequence. The human motion is encoded per frame as $\mathbf{m}_t$, made of the horizontal root velocity, the 6D root orientation, and the root-relative joint positions and velocities. The robot motion $\mathbf{m}^{\mathrm{bot}}_t$ additionally contains the joint angles. A 1D residual convolutional encoder with temporal downsampling feeds a transformer with full, non-causal self-attention, and an upsampling convolutional decoder predicts the robot sequence. The network is trained with

$$\mathcal{L}_{\mathrm{NMR}} = \sum_{t=1}^{T} \left\|\mathbf{m}^{\mathrm{bot}}_t - \hat{\mathbf{m}}^{\mathrm{bot}}_t\right\|_1, \qquad \hat{\mathbf{m}}^{\mathrm{bot}}_{1:T} = F_\psi(\mathbf{m}_{1:T}). \tag{15}$$

The training pairs are produced in three stages. Human clips are first filtered for excessive jerk, unsupported centre of mass, and insufficient foot contact. The remainder are retargeted with GMR and filtered again by joint velocity, self-intersection, and foot clearance. The resulting motions are then clustered, one reinforcement-learning tracking policy is trained per cluster in simulation, and the policy rollouts provide about 30k physically executed human–robot pairs. F_ψ is pre-trained for 500 epochs on the kinematic retargeting data and fine-tuned for 50 epochs on the rollout pairs. At inference NMR is a single forward pass with no optimization or post-processing, so physical plausibility rests entirely on what the rollout data taught the network.

Algorithm 4 Retarget Robot Motion (NMR)

Require: Trained network F_ψ, human motion features $\mathbf{m}_{1:T}$
Ensure: $\mathbf{q}_{1:T}$
1: $\hat{\mathbf{m}}^{\mathrm{bot}}_{1:T} \leftarrow F_\psi(\mathbf{m}_{1:T})$ ▷ single forward pass over the entire sequence
2: $\mathbf{q}_{1:T} \leftarrow$ root pose and joint angles read from $\hat{\mathbf{m}}^{\mathrm{bot}}_{1:T}$

C.4 Evaluation Metrics

This section gives the exact form of the metrics summarized in Section 4. All test sequences are evaluated at 30 Hz under two fixed protocols applied identically to every method, so no baseline is scored under an alignment, scale fit, or contact definition derived from its own output.

Kinematic fidelity. Fidelity is measured on twelve GMR keypoints recovered directly from the raw source SMPL-X sequence, with no method-dependent alignment or scale fit. Human keypoints are taken relative to the pelvis and rescaled by the morphology convention of GMR [Araujo et al. (2025)], a fixed height ratio of 1.66/1.80 times 0.9 on the six lower-body and 0.8 on the six upper-body keypoints. We use this per-group scaling because the G1 does not share the limb proportions of the source subject, and we apply it identically to every method. Robot keypoints are taken root-

relative from forward kinematics. Both sequences are then yaw-aligned by their own heading. Writing $\Delta^{\mathrm{loc}}_{t,j}$ and $\Delta^{\mathrm{wld}}_{t,j}$ for the resulting per-keypoint discrepancies in the root-local frame and in a world frame anchored at the first frame,

$$\begin{aligned} \mathrm{MPKPE} &= \mathrm{mean}_{t,j}\,\|\Delta^{\mathrm{loc}}_{t,j}\|_2, & \text{W-MPKPE} &= \mathrm{mean}_{t,j}\,\|\Delta^{\mathrm{wld}}_{t,j}\|_2, \\ \mathrm{MPKVE} &= \mathrm{mean}_{t,j}\,\|\dot{\Delta}^{\mathrm{loc}}_{t,j}\|_2, & \mathrm{PCK@10} &= \mathrm{mean}_{t,j}\,\mathbb{1}\big[\|\Delta^{\mathrm{loc}}_{t,j}\|_2 \le 10\,\mathrm{cm}\big], \end{aligned} \tag{16}$$

where $\dot{\Delta}^{\mathrm{loc}}_{t,j}$ is the discrepancy between the robot and human keypoint velocities, each obtained by first-order finite differences of the root-local positions at 30 Hz, and $\mathbb{1}[\cdot]$ is the indicator function. The **End** column applies the MPKPE expression to the endpoint keypoint subgroup only. Lower- and upper-body subgroups are likewise reported in the released evaluation logs.

Motion quality. Every quality metric is anchored to the contacts of the *source human*, never to contacts the evaluated method declares itself, so all methods are judged on the same frames. Foot skating and floating are evaluated on the two feet, and penetration and contact preservation over the whole body, on the $G = 12$ groups of Table 5. Source contact is resampled to the robot frame rate by nearest-neighbour indexing, and all velocities are first-order finite differences at 30 Hz.

Foot metrics. The source foot-contact mask gives, for each frame t and foot $f \in \{\mathrm{L}, \mathrm{R}\}$, an indicator $s_{t,f}$, and we write

$$\mathcal{S} = \{(t, f) : s_{t,f} = 1\} \tag{17}$$

for the resulting set of source-contact frame–foot samples. Sole geometry is evaluated by forward kinematics. Each sole sphere contributes its centre height minus its radius, and for foot f we take the tangential speed $v_{t,f}$ as the mean horizontal sphere speed and the sole height $h_{t,f}$ as the minimum sphere height above the ground plane. The skating and floating columns of Table 1(b) are the fraction of $\mathcal{S}$ on which the violation holds and its extreme magnitude over the same set,

$$\begin{aligned} \mathrm{Skate}^{\mathrm{dur}} &= \frac{|\{(t,f) \in \mathcal{S} : v_{t,f} > v_{\mathrm{slide}}\}|}{|\mathcal{S}|}, & \mathrm{Skate}^{\mathrm{max}} &= \max_{(t,f)\in\mathcal{S}} v_{t,f}, \\ \mathrm{Float}^{\mathrm{dur}} &= \frac{|\{(t,f) \in \mathcal{S} : h_{t,f} > \epsilon_{\mathrm{float}}\}|}{|\mathcal{S}|}, & \mathrm{Float}^{\mathrm{max}} &= \max_{(t,f)\in\mathcal{S}} \max(0, h_{t,f}). \end{aligned} \tag{18}$$

Whole-body metrics. The source schedule is a dense contact map on the 2,256 SMPL-X template vertices of Appendix B.1, computed geometrically from the source human motion alone, independently of any retargeting method and of the contact predictor, and thresholded at 0.5. Group g is in source contact at frame t, $s_{t,g} = 1$, when at least three of its vertices are in contact, and its source contact events $\mathcal{E}_g$ are the maximal runs of such frames lasting at least 0.10 s. On the robot, with $z_{t,n}$ the height of atlas node n above the ground plane, group g realizes contact at frame t, $r_{t,g} = 1$, when at least three of its nodes satisfy $|z_{t,n}| \le d_{\mathrm{c}}$. Writing $\mathcal{S}_{\mathrm{wb}} = \{(t, g) : s_{t,g} = 1\}$ for the source-contact group–frame samples and $\pi_{t,g} = \max_{n\in g} \max(0, -z_{t,n})$ for the deepest penetration of group g at frame t, the penetration columns are

$$\mathrm{Pen}^{\mathrm{dur}} = \frac{|\{(t,g) \in \mathcal{S}_{\mathrm{wb}} : \pi_{t,g} > \epsilon_{\mathrm{pen}}\}|}{|\mathcal{S}_{\mathrm{wb}}|}, \qquad \mathrm{Pen}^{\mathrm{max}} = \max_{(t,g)\in\mathcal{S}_{\mathrm{wb}}} \pi_{t,g}. \tag{19}$$

An event $e \in \mathcal{E}_g$ is realized when $r_{t,g} = 1$ on at least half of its frames, and contact preservation is the event recall averaged over the groups Γ_{src} that have at least one source event,

$$\mathrm{ContactPres} = \frac{1}{|\Gamma_{\mathrm{src}}|} \sum_{g\in\Gamma_{\mathrm{src}}} \frac{|\{e \in \mathcal{E}_g : \mathrm{mean}_{t\in e}\, r_{t,g} \ge 0.5\}|}{|\mathcal{E}_g|}, \tag{20}$$

so that frequent foot contacts do not dominate rarer hand, knee, pelvis, or forearm contacts.

Joint limits. The joint-limit column is a per-sample rate over all T frames and 29 joints, not a per-frame rate,

$$\mathrm{JointLim} = \frac{1}{29\,T} \left|\left\{(t,j) : \theta_{t,j} < \theta^{\min}_j - \epsilon_\theta \ \text{ or } \ \theta_{t,j} > \theta^{\max}_j + \epsilon_\theta\right\}\right|, \tag{21}$$

with $\theta^{\min}_j, \theta^{\max}_j$ the hardware limits of joint j.

The thresholds are $v_{\text{slide}} = 0.10\,\text{m/s}$, $\epsilon_{\text{float}} = 8\,\text{cm}$, $\epsilon_{\text{pen}} = 1\,\text{cm}$, $d_{\text{c}} = 2.5\,\text{cm}$, and $\epsilon_\theta = 10^{-5}\,\text{rad}$, so the joint-limit test is effectively exact. Rates are computed per sequence and then averaged across the test sequences, which is why a method can report a zero mean duration together with a nonzero peak magnitude. The LAFAN1 results of Table 2 and Appendix D.1 use the same evaluators with $v_{\text{slide}} = 0.30\,\text{m/s}$, the threshold under which the released Unitree-LAFAN references are reported.

Downstream tracking. Each policy is evaluated on the same test set, grouped into short, medium, and long duration bins of 195, 298, and 170 sequences. Success and tracking errors are first obtained for each sequence with the tracker's own termination criteria and tracking-error metrics, following the default criteria of its public implementation without modification (Appendix C.5), and are then aggregated over sequences. **SR** is the fraction of sequences the policy tracks to completion without falling or triggering early termination. **MPJPE** is the mean Euclidean error between the robot's joint positions and those of the reference, taken root-relative, and **W-MPJPE** is the same quantity in a world frame, so that it also reflects global drift. Both are averaged over the sequences of each bin and reported in meters.

C.5 Downstream Control Framework

Downstream control is evaluated with two separately implemented reinforcement-learning tracking pipelines, described below. Each is trained with the default Unitree G1 29-DoF whole-body tracking configuration of its public implementation, and no observation, action, reward, randomization, termination, or training setting is changed. Within a pipeline, the only input that differs between runs is the reference set. Each policy is trained on the 6,000-sequence training split of Appendix C.2, retargeted by the method under evaluation, and is evaluated on the held-out test set, so no evaluation reference is seen during policy training. Every policy is trained from scratch with the default random seed of its public implementation, and all results are reported for the checkpoint after 12,000 training iterations.

C.5.1 BeyondMimic Tracker

We use the official BeyondMimic implementation [Liao et al. (2026)], built on Isaac Lab, with physics simulated at 200 Hz and the policy running at 50 Hz.

Observations and actions. The torso is the anchor body. The actor observes the reference joint positions and velocities, the pose error of the anchor, the base linear and angular velocities in the base frame, the joint positions and velocities, and the previous action. The anchor pose error is the reference anchor position and orientation expressed in the robot's anchor frame, with the orientation encoded by the first two columns of its rotation matrix. Uniform noise is added to every term except the reference joint command and the previous action. The critic receives the same terms without noise, together with the positions and orientations of the tracked bodies relative to the robot's anchor. The action $\mathbf{a}_t$ sets joint position targets $\bar{\mathbf{q}}_j + \alpha_j a_{t,j}$ around a nominal configuration $\bar{\mathbf{q}}$, with $\alpha_j = 0.25\,\tau_j^{\max}/K_j^p$, tracked by a PD controller whose gains $K_j^p = I_j\omega_n^2$ and $K_j^d = 2I_j\zeta\omega_n$ follow from the reflected actuator inertia I_j, a natural frequency of 10 Hz, and a damping ratio $\zeta = 2$. Fourteen bodies are tracked, namely the pelvis, the torso, and the left and right hip roll, knee, ankle roll, shoulder roll, elbow, and wrist yaw links.

Rewards. Six tracking terms have the form $\exp(-\|\mathbf{e}\|^2/\sigma^2)$, with $\mathbf{e}$ the error between reference and robot. They track the global position ($w = 0.5$, $\sigma = 0.3\,\text{m}$) and orientation ($w = 0.5$, $\sigma = 0.4\,\text{rad}$) of the torso anchor and, averaged over the tracked bodies, their anchor-relative positions ($w = 1.0$, $\sigma = 0.3\,\text{m}$) and orientations ($w = 1.0$, $\sigma = 0.4\,\text{rad}$) and their linear ($w = 1.0$, $\sigma = 1.0\,\text{m/s}$) and angular ($w = 1.0$, $\sigma = 3.14\,\text{rad/s}$) velocities. Three regularizers penalize the squared action rate ($w = -0.1$), joint positions outside soft limits set at 90% of the hardware range ($w = -10$), and contact forces above 1 N on any body other than the ankles and wrists ($w = -0.1$).

Randomization and initialization. At start-up, the static and dynamic friction ([0.3,1.6], [0.3,1.2]) and restitution ([0,0.5]) of the robot links, the torso centre-of-mass offset ($\pm 2.5\,\text{cm}$ along x, $\pm 5\,\text{cm}$ along y and z), and the nominal joint configuration $\bar{\mathbf{q}}$ ($\pm 0.01\,\text{rad}$, affecting both actions

and observations) are randomized, and velocity pushes of up to 0.5 m/s horizontally, 0.2 m/s vertically, and 0.52, 0.52, and 0.78 rad/s in roll, pitch, and yaw are applied every 1–3 s. Episodes last at most 10 s. At reset the start frame is drawn by adaptive sampling that favours temporal bins with recent failures, and the robot is initialized at the reference state perturbed by up to ± 0.05 m in horizontal position, ± 0.01 m in height, ± 0.1 rad in roll and pitch, ± 0.2 rad in yaw, ± 0.1 rad in joint positions, and the push ranges in root velocity.

Termination. An episode terminates when the torso anchor height deviates from the reference by more than 0.25 m, when the z-components of the gravity direction projected into the reference and robot torso frames differ by more than 0.8, when the height of either ankle or wrist deviates from the reference by more than 0.25 m, or at timeout.

Training. PPO runs with 4096 parallel environments and 24 steps per environment. Actor and critic are MLPs with hidden layers [512,256,128] and ELU activations, with empirical observation normalization. The learning rate is 10^{-3} with an adaptive schedule targeting a KL divergence of 0.01, with $\gamma = 0.99$, $\lambda = 0.95$, clip parameter 0.2, entropy coefficient 0.005, 5 learning epochs, and 4 mini-batches per update.

C.5.2 HOLOSOMA TRACKER

We use the whole-body tracking task of Holosoma [Amazon FAR (2025)], the open-source training framework released as part of the OmniRetarget project [Yang et al. (2025)], with its default PPO preset for the 29-DoF G1. It runs in Isaac Sim with physics at 200 Hz and the policy at 50 Hz. We use the released default configuration described below, which differs in some terms from the minimal setting reported in the OmniRetarget paper (for example, it uses the torso instead of the pelvis as the anchor and keeps the full BeyondMimic reward).

Observations and actions. The actor is proprioceptive. It observes the reference joint positions and velocities, the orientation of the reference torso anchor in the robot's anchor frame, the base angular velocity, the joint positions and velocities, and the previous action, with additive noise, but neither the anchor position error nor the base linear velocity. The critic additionally receives, without noise, the anchor position error, the base linear velocity, and the positions and orientations of the tracked bodies relative to the robot's anchor. The torso is again the anchor body, and the same fourteen bodies as above are tracked. Actions are joint position targets around the default configuration with the same per-joint scaling $0.25\,\tau_j^{\max}/K_j^p$ and the same PD gains as the BeyondMimic tracker, and self-collisions are enabled in simulation.

Rewards. The default reward uses the same nine terms, weights, and kernel widths as the BeyondMimic tracker, namely anchor position and orientation tracking, anchor-relative body position and orientation tracking, body linear and angular velocity tracking, and penalties on action rate, joint positions outside soft limits at 90% of the hardware range, and contact forces above 1 N on bodies other than the ankles, wrists, and foot contact points.

Randomization and initialization. At start-up, link friction and restitution, the torso centre-of-mass offset, and a joint-position bias are randomized over the same ranges as above, and velocity pushes with the same magnitudes are applied every 1–3 s. Actuator-gain and action-delay randomization are available in the framework but disabled by default. Episodes last at most 10 s, the start frame is drawn by adaptive sampling, and the initial state is perturbed with the same ranges as above.

Termination. An episode terminates when the torso anchor height deviates from the reference by more than 0.5 m, when the projected-gravity tilt error of the anchor exceeds 0.8, when the height of either ankle or wrist deviates from the reference by more than 0.25 m, or at timeout.

Training. PPO runs with 4096 parallel environments and 24 steps per environment. Actor and critic are MLPs with hidden layers [512,256,128] and ELU activations, with empirical observation normalization. Actor and critic use separate optimizers with learning rate 10^{-3}, and the remaining settings are $\gamma = 0.99$, $\lambda = 0.95$, clip parameter 0.2, entropy coefficient 0.005, adaptive schedule

Table 7: Full LAFAN1 comparison. Robot-side motion quality of NeCR, the released Unitree-LAFAN references, and all retargeting baselines on the same LAFAN1 sequences. This table extends Table 2(a) of the main text. Penetration and contact preservation are whole-body, and skating and floating are the foot metrics of Appendix C.4. Values are mean ± standard deviation over sequences, and first and second rank within each column.

	Foot skating		Penetration		Floating		Contact pres.
Method	**Dur. (%)↓**	**Max (cm/s)↓**	**Dur. (%)↓**	**Max (cm)↓**	**Dur. (%)↓**	**Max (cm)↓**	**(%)↑**
Unitree-LAFAN	12.46 ± 7.03	$\mathbf{405.3 \pm 174.0}$	21.06 ± 11.69	4.70 ± 4.49	0.03 ± 0.10	5.42 ± 2.05	77.47 ± 22.80
GMR	15.40 ± 8.88	518.4 ± 220.4	59.64 ± 16.92	10.38 ± 3.83	$\mathbf{0.00 \pm 0.00}$	$\mathbf{1.10 \pm 1.28}$	$\mathbf{80.23 \pm 21.13}$
NMR	24.66 ± 14.93	445.3 ± 188.2	49.12 ± 23.69	10.10 ± 8.17	0.12 ± 0.99	4.80 ± 2.98	66.63 ± 36.75
OmniRetarget	$\mathbf{9.21 \pm 5.50}$	433.6 ± 192.3	0.03 ± 0.15	0.20 ± 0.07	0.07 ± 0.25	5.88 ± 2.88	46.85 ± 29.51
NeCR (Ours)	11.88 ± 6.55	442.5 ± 184.4	$\mathbf{0.00 \pm 0.00}$	$\mathbf{0.08 \pm 0.04}$	0.01 ± 0.09	3.49 ± 1.83	76.48 ± 29.93

with KL target 0.01, 5 learning epochs, and 4 mini-batches. Episodes are initialized at random lengths at the start of training.

C.5.3 Evaluation

Both pipelines are evaluated with their own termination criteria and tracking-error metrics, unmodified. Each test sequence is evaluated separately. It counts as tracked when it reaches its end without triggering any termination other than the timeout, and its MPJPE and W-MPJPE are recorded. The per-sequence results are then averaged within each duration bin and over all sequences.

D Additional Results

D.1 Full LAFAN1 Comparison

Table 7 extends Table 2(a) of the main text with GMR, NMR, and OmniRetarget on the same LAFAN1 sequences and the same 0.3 m/s skating threshold. On the whole-body columns, the baselines fail in one of two opposite ways. GMR, NMR, and the released references reach the contact surface by passing through it. They preserve many source contacts, GMR the most of any method, but at the price of deep penetration, so contact preservation alone cannot distinguish a realized contact from an overshot one. OmniRetarget fails in the opposite direction. It stops above the surface, and the gap drifts from frame to frame, so it almost never penetrates but realizes fewer than half of the source contact events. NeCR is the only method that avoids both failure modes. It has no penetrating sample and preserves contact at the level of the released references. On the foot columns, OmniRetarget keeps the lowest skating duration and NeCR the second lowest.

D.2 Full Downstream Tracking Results

We additionally evaluate policy robustness under a policy-by-reference cross-validation protocol. Each row in Table 8 denotes the reference distribution used to train the BeyondMimic tracker, while each column denotes the reference distribution used at evaluation time. Thus, the diagonal entries are matched-reference evaluations and the off-diagonal entries measure transfer across retargeting methods. The table reports the success count out of the 195/298/170 sequences in the short/medium/long bins, followed by the mean ± standard deviation of MPJPE and W-MPJPE in meters. The corresponding Holosoma tracker results are reported in Table 9.

Three patterns hold on both trackers. The long bin is the hardest for every policy and every reference. Policies trained on one baseline transfer poorly to the references of another, most visibly the NMR-trained policies on OmniRetarget references and the OmniRetarget-trained Holosoma policy on GMR references. NeCR-trained policies rank first on every evaluation reference with the Holosoma tracker and on all but the OmniRetarget references with the BeyondMimic tracker, where only the matched OmniRetarget policy is better.

Table 8: BeyondMimic tracker policy-by-reference cross-validation on the held-out test set. The success-rate results are split into four compact panels, one for each evaluation reference, with Short, Medium, Long, and All shown horizontally. The tracking-error results use grouped columns for each evaluation reference and report MPJPE and W-MPJPE side by side. Success cells show the successful-sequence count and percentage. Green and yellow cells mark the best and second-best results, respectively, within each panel or metric and evaluation-reference column.

(a) Evaluation reference: GMR

Policy trained on	Short	Medium	Long	All
GMR	107/195 (54.87%)	203/298 (68.12%)	54/170 (31.76%)	364/663 (54.90%)
NMR	115/195 (58.97%)	229/298 (76.85%)	78/170 (45.88%)	422/663 (63.65%)
OmniRetarget	99/195 (50.77%)	177/298 (59.40%)	62/170 (36.47%)	338/663 (50.98%)
NeCR	**148/195 (75.90%)**	**259/298 (86.91%)**	**99/170 (58.24%)**	**506/663 (76.32%)**

(b) Evaluation reference: NMR

Policy trained on	Short	Medium	Long	All
GMR	152/195 (77.95%)	250/298 (83.89%)	80/170 (47.06%)	482/663 (72.70%)
NMR	173/195 (88.72%)	281/298 (94.30%)	131/170 (77.06%)	585/663 (88.24%)
OmniRetarget	154/195 (78.97%)	270/298 (90.60%)	123/170 (72.35%)	547/663 (82.50%)
NeCR	**183/195 (93.85%)**	**289/298 (96.98%)**	**150/170 (88.24%)**	**622/663 (93.82%)**

(c) Evaluation reference: OmniRetarget

Policy trained on	Short	Medium	Long	All
GMR	121/195 (62.05%)	158/298 (53.02%)	33/170 (19.41%)	312/663 (47.06%)
NMR	93/195 (47.69%)	76/298 (25.50%)	16/170 (9.41%)	185/663 (27.90%)
OmniRetarget	167/195 (85.64%)	**280/298 (93.96%)**	**122/170 (71.76%)**	**569/663 (85.82%)**
NeCR	**176/195 (90.26%)**	266/298 (89.26%)	96/170 (56.47%)	538/663 (81.15%)

(d) Evaluation reference: NeCR

Policy trained on	Short	Medium	Long	All
GMR	140/195 (71.79%)	239/298 (80.20%)	65/170 (38.24%)	444/663 (66.97%)
NMR	145/195 (74.36%)	224/298 (75.17%)	74/170 (43.53%)	443/663 (66.82%)
OmniRetarget	134/195 (68.72%)	259/298 (86.91%)	98/170 (57.65%)	491/663 (74.06%)
NeCR	**190/195 (97.44%)**	**296/298 (99.33%)**	**154/170 (90.59%)**	**640/663 (96.53%)**

(e) Tracking error (m) ↓

Policy trained on	GMR		NMR		OmniRetarget		NeCR	
	MPJPE	W-MPJPE	MPJPE	W-MPJPE	MPJPE	W-MPJPE	MPJPE	W-MPJPE
GMR	0.0773 ± 0.0304	0.2753 ± 0.3008	0.0693 ± 0.0248	0.2263 ± 0.1898	0.0865 ± 0.0197	0.3360 ± 0.2697	0.0748 ± 0.0238	0.3328 ± 0.2975
NMR	0.0696 ± 0.0295	0.2400 ± 0.2294	0.0561 ± 0.0283	0.1897 ± 0.1955	0.0737 ± 0.0260	0.2395 ± 0.2142	0.0609 ± 0.0239	0.2745 ± 0.2286
OmniRetarget	0.0675 ± 0.0257	$\mathbf{0.1864 \pm 0.1961}$	0.0551 ± 0.0227	$\mathbf{0.1762 \pm 0.1709}$	$\mathbf{0.0500 \pm 0.0179}$	$\mathbf{0.1567 \pm 0.1562}$	0.0550 ± 0.0190	$\mathbf{0.1961 \pm 0.1988}$
NeCR	$\mathbf{0.0557 \pm 0.0244}$	0.2339 ± 0.2264	$\mathbf{0.0435 \pm 0.0220}$	0.1894 ± 0.2087	0.0538 ± 0.0223	0.2402 ± 0.1922	$\mathbf{0.0404 \pm 0.0152}$	0.2075 ± 0.2308

Table 9: Holosoma tracker policy-by-reference cross-validation at the 12k checkpoint on the held-out test set. Success-rate results are split into four compact panels, one for each evaluation reference, with Short, Medium, Long, and All shown horizontally. Tracking-error results report MPJPE and W-MPJPE in meters with mean $\pm$ standard deviation. Green and yellow cells mark the best and second-best results, respectively.

(a) Evaluation reference: GMR

Policy trained on	Short	Medium	Long	All
GMR	98/195 (50.26%)	174/298 (58.39%)	24/170 (14.12%)	296/663 (44.65%)
NMR	123/195 (63.08%)	200/298 (67.11%)	58/170 (34.12%)	381/663 (57.47%)
OmniRetarget	37/195 (18.97%)	65/298 (21.81%)	8/170 (4.71%)	110/663 (16.59%)
NeCR	**156/195 (80.00%)**	**254/298 (85.23%)**	**92/170 (54.12%)**	**502/663 (75.72%)**

(b) Evaluation reference: NMR

Policy trained on	Short	Medium	Long	All
GMR	120/195 (61.54%)	218/298 (73.15%)	47/170 (27.65%)	385/663 (58.07%)
NMR	165/195 (84.62%)	272/298 (91.28%)	114/170 (67.06%)	551/663 (83.11%)
OmniRetarget	111/195 (56.92%)	191/298 (64.09%)	32/170 (18.82%)	334/663 (50.38%)
NeCR	**177/195 (90.77%)**	**286/298 (95.97%)**	**147/170 (86.47%)**	**610/663 (92.01%)**

(c) Evaluation reference: OmniRetarget

Policy trained on	Short	Medium	Long	All
GMR	81/195 (41.54%)	117/298 (39.26%)	10/170 (5.88%)	208/663 (31.37%)
NMR	73/195 (37.44%)	42/298 (14.09%)	6/170 (3.53%)	121/663 (18.25%)
OmniRetarget	136/195 (69.74%)	**240/298 (80.54%)**	**74/170 (43.53%)**	450/663 (67.87%)
NeCR	**148/195 (75.90%)**	**240/298 (80.54%)**	71/170 (41.76%)	**459/663 (69.23%)**

(d) Evaluation reference: NeCR

Policy trained on	Short	Medium	Long	All
GMR	105/195 (53.85%)	195/298 (65.44%)	37/170 (21.76%)	337/663 (50.83%)
NMR	122/195 (62.56%)	206/298 (69.13%)	68/170 (40.00%)	396/663 (59.73%)
OmniRetarget	75/195 (38.46%)	154/298 (51.68%)	26/170 (15.29%)	255/663 (38.46%)
NeCR	**177/195 (90.77%)**	**291/298 (97.65%)**	**144/170 (84.71%)**	**612/663 (92.31%)**

(e) Tracking error (m) ↓

Policy trained on	GMR		NMR		OmniRetarget		NeCR	
	MPJPE	W-MPJPE	MPJPE	W-MPJPE	MPJPE	W-MPJPE	MPJPE	W-MPJPE
GMR	0.0896 ± 0.0257	0.6579 ± 0.6761	0.0786 ± 0.0200	0.6135 ± 0.7169	0.1052 ± 0.0231	0.7504 ± 0.7565	0.0851 ± 0.0194	0.7871 ± 0.9616
NMR	0.0716 ± 0.0284	$\mathbf{0.6483 \pm 0.7487}$	0.0536 ± 0.0176	0.5545 ± 0.5582	0.0880 ± 0.0229	0.8099 ± 1.1523	0.0647 ± 0.0176	1.0334 ± 1.3024
OmniRetarget	0.0941 ± 0.0205	0.6495 ± 0.6856	0.0825 ± 0.0156	0.9837 ± 0.8992	0.0765 ± 0.0203	$\mathbf{0.5733 \pm 0.5704}$	0.0876 ± 0.0181	0.8810 ± 0.8841
NeCR	$\mathbf{0.0662 \pm 0.0281}$	0.6765 ± 0.7681	$\mathbf{0.0508 \pm 0.0145}$	$\mathbf{0.5007 \pm 0.4828}$	$\mathbf{0.0707 \pm 0.0225}$	1.1804 ± 1.2346	$\mathbf{0.0514 \pm 0.0139}$	$\mathbf{0.5446 \pm 0.5287}$

D.3 Qualitative Results

Figure 8 widens the single-sequence comparison of Figure 3 to ten source motions, covering ordinary locomotion, whole-body dynamic motion, and the low-height postures that motivate this work. Every method is shown on the same frame of the same source sequence, rendered on the Unitree G1. NeCR is the only method without a marked failure on any of the ten motions.

The baselines fail in three ways. The first is out-of-distribution collapse. On the cartwheel, fall, and cross-legged sitting rows, NMR returns a pose with no recognisable correspondence to the source. These postures are far from the locomotion-dominated data it is trained on, since its training clips are filtered for insufficient foot contact (Appendix C.3), and a learned mapping is only as reliable as the motions it was trained on. NMR also inherits the limits of the policy used to refine its targets. A motion the policy cannot track is refined toward one it can, so dynamic motions such as the jump come out with reduced amplitude. The second failure is a contact artifact. GMR penetrates the floor

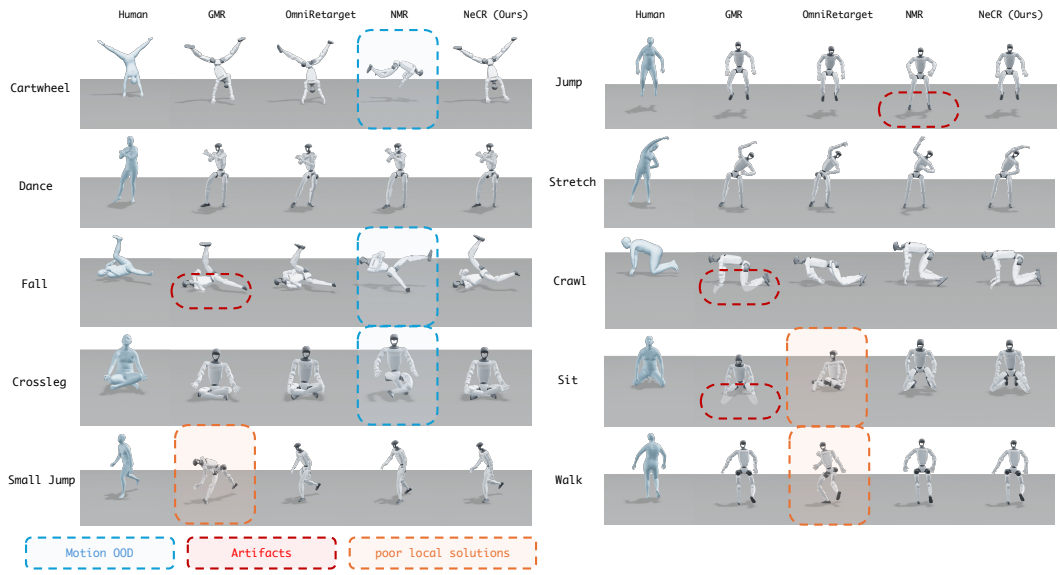

Figure 8: Qualitative comparison across ten motion types. Each row is one source motion, and the columns show the SMPL-X source and the result of each method at the same frame. Blue boxes mark out-of-distribution collapse, red boxes contact artifacts, and orange boxes poor local solutions.

on the fall, crawl, and sit rows, and NMR on the jump row, because neither method corrects contact after mapping the pose. The third is a poor local solution. On the small-jump row for GMR and the sit and walk rows for OmniRetarget, the optimizer converges and reports success, but the whole pose has drifted away from the source. The solution is kinematically valid, so the optimizer cannot detect this failure on its own.

NeCR avoids the three failures for different reasons. Its nominal retargeter is learned as well and carries the same risk of collapse, which is why it is trained on BONES-SEED motions selected to span a wide range of motion types (Appendix B.2). An aggregate mean over the test set understates how badly a mapping fails on such tail motions. Its targets are not refined by a policy, so no policy bias enters the reference. Contact artifacts are removed by CAP. Poor local solutions do not occur because CAP starts from the nominal pose, which is already close to the source, and moves only the support-relevant DoFs by a small amount, so the optimizer never leaves the neighbourhood of the source pose.

Figure 9 shows a failure that a single frame does not capture. Some reduction in global displacement is expected for every method, since the G1 is smaller than the human and covers less ground per stride. Beyond this, NMR is the only method that shrinks the motion, with a shorter path than the other methods and an incomplete rise. Its targets are refined by a tracking policy, so the reference inherits what the policy could execute rather than what the human did. The other methods, including NeCR, reproduce both phases in full. MPKVE measures the same effect in aggregate, and NeCR has the lowest MPKVE of the four methods in Table 1(a).

D.4 Inference Speed

Table 2(b) reports the per-frame breakdown on LAFAN1. The corresponding wall-clock totals over the full set are 7.78 s for feature extraction, 25.78 s for the nominal retargeter, 297.75 s for contact-mode prediction and event compilation, and 1,116.35 s for contact-aware projection with three Gauss–Newton sweeps, 1,447.65 s in total. All timings are single-threaded. One Python process handles the sequences serially on an Intel Core Ultra 9 285K CPU (24 cores, no hyper-threading) and an NVIDIA RTX 5080 GPU, with no batching across sequences. Sequences are independent, so throughput scales with the number of parallel workers, and the figures above are the unparallelized ones. Both networks are trained on BONES-SEED only and are applied to LAFAN1 without retraining or fine-tuning.

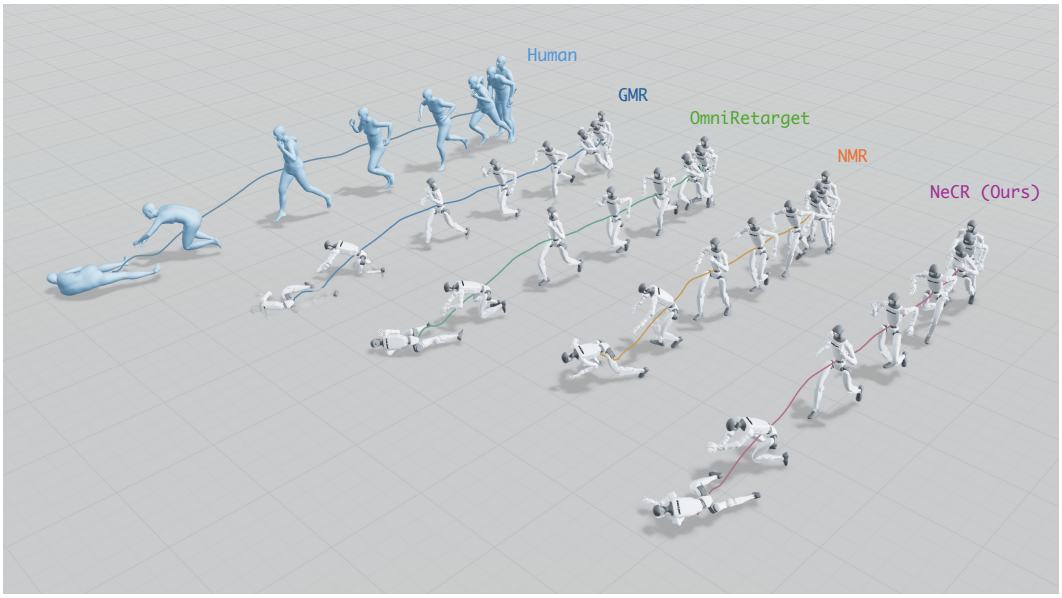

Figure 9: Retargeting a long sequence that starts prone, pushes up, and breaks into a run. Each lane shows one method on the same source, with the curve tracing the root path and the poses sampled at the same source frames.

E Ablation Studies

NeCR has two components beyond the nominal retargeter, the temporal contact mode predicted by g_ϕ and the contact-aware projection CAP. The ablations in this appendix test each of them. Every experiment changes one element and keeps the nominal trajectory and everything else fixed.

For the contact mode, we ask whether the projector needs a learned and continuous specification. Appendix E.1 replaces the learned modes with heuristic height and velocity cues, and shows that the learned modes give better contact evidence at the same fidelity. Appendix E.2 replaces the continuous confidence with binary weights, and shows that weighting the correction by confidence prevents over-correction of uncertain contacts.

For CAP, we ask whether its design is minimal and its cost is low. Appendix E.3 removes one objective term at a time, and shows that every term is necessary for contact consistency. Appendix E.4 varies the number of Gauss–Newton sweeps, and shows that one sweep already removes the contact artifacts and three sweeps suffice for convergence.

E.1 Contact Estimator

Table 10 isolates the contact estimator. The nominal motion, compiler, and CAP optimizer are fixed, and only the contact evidence handed to them changes. Full NeCR uses the learned temporal contact modes. The height/velocity variant replaces them with thresholded height and velocity cues over all body groups, and the surface-probability variant uses the nominal surface likelihoods directly.

Penetration is near zero for all three variants because they share the same projection, so the differences come only from the contact evidence. Both substitutes trade contact quality for a small gain in fidelity. The height/velocity heuristic has the best pose fidelity, but by margins of at most 0.2 cm, while its skating duration rises by half and both its peak sliding speed and whole-body contact preservation degrade. The raw surface probabilities sit in between. They recover most of the skating duration but not the peak sliding speed, and they preserve the least contact. The learned modes therefore provide the best temporal contact evidence for local correction, at a fidelity cost of 0.07 cm MPKPE, about 1% of the error.

Table 10: Contact-estimator ablation. Only the contact evidence handed to the projector changes, while the nominal motion, compiler, optimizer, and evaluation protocol are fixed. Values are mean $\pm$ standard deviation over sequences, and highlighted cells mark the best variant in each column.

(a) Kinematic fidelity

Variant	MPKPE (cm)↓	W-MPKPE (cm)↓	MPKVE (cm/s)↓	End (cm)↓	PCK@10 (%)↑
Full NeCR (learned modes)	7.018 ± 0.508	13.220 ± 12.811	11.962 ± 12.762	4.708 ± 0.984	84.720 ± 4.482
Whole-body height/velocity heuristic	**6.949 ± 0.529**	**13.081 ± 12.772**	11.898 ± 13.126	**4.518 ± 1.036**	**84.761 ± 4.556**
Nominal surface probability	6.976 ± 0.524	13.122 ± 12.748	**11.864 ± 13.083**	4.599 ± 1.014	84.711 ± 4.519

(b) Contact quality

	Foot skating		Penetration		Floating		Contact pres.
Variant	Dur. (%)↓	Max (cm/s)↓	Dur. (%)↓	Max (cm)↓	Dur. (%)↓	Max (cm)↓	(%)↑
Full NeCR (learned modes)	**4.39 ± 5.83**	**77.85 ± 83.20**	**0.000 ± 0.000**	**0.010 ± 0.010**	**0.14 ± 1.18**	**3.29 ± 6.42**	**94.09 ± 17.42**
Whole-body height/velocity heuristic	6.60 ± 8.73	92.23 ± 76.07	0.037 ± 0.801	0.040 ± 0.476	0.20 ± 1.33	3.89 ± 6.41	92.44 ± 18.94
Nominal surface probability	4.91 ± 6.95	89.50 ± 79.41	0.035 ± 0.868	0.025 ± 0.317	0.19 ± 1.30	3.94 ± 6.40	92.28 ± 19.09

E.2 Effect of Binarizing Contact Weights

The temporal contact mode is continuous, whereas prior pipelines supply contact as a binary label. This ablation tests whether the continuous confidence matters. The estimator ablation replaces the evidence, whereas this one replaces only its form. Keeping the compiled events, intervals, patches, anchors, hard constraints, and every global optimizer coefficient fixed, we set each positive optimizer-facing weight to one, which is what a binary contact label supplies. Tables 11 and 12 report the result over the 662 motions both variants solve, with cluster-bootstrap confidence intervals over 10,000 resamples of motions.

Binarization keeps the same contact events but applies full weight to every frame, node, and stick term within them, regardless of the predicted confidence. Frames near contact onset and release, nodes at the edge of a patch, and feet that roll or slide in the source are then corrected as strongly as a certain contact. As a result, support-phase sliding increases from 4.39% to 15.06% (95% CI of the increase [9.24, 12.16]), penetration appears where there was none (CI [0.07, 0.17]), and MPKPE increases by 0.20 cm (CI [0.18, 0.22]). The corrections also become larger in root translation, root rotation, and joint rotation, and one motion becomes infeasible for the coupled QP. Whole-body contact preservation does not change significantly, with a confidence interval of the difference that crosses zero ($[-0.44, 0.12]$). Therefore, continuous weights do not realize more contacts than binary weights. Their benefit is that the correction scales with confidence, so a contact with weak evidence costs $O(w)$ rather than $O(1)$ (Section 3.4).

Table 11: Continuous versus binary contact weights, contact quality. The binary variant replaces every positive optimizer-facing weight w_t by $\mathbf{1}[w_t > 0]$ after event compilation, leaving the event structure, intervals, patches, anchors, hard constraints, and every global optimizer coefficient unchanged. Values are paired over the 662 motions both variants solve and follow the protocol of Table 1(b).

	Foot skating		Penetration		Floating		Contact pres.
Weighting	Dur. (%)↓	Max (cm/s)↓	Dur. (%)↓	Max (cm)↓	Dur. (%)↓	Max (cm)↓	(%)↑
Continuous (ours)	**4.386 ± 5.831**	**77.87 ± 83.20**	**0.000 ± 0.000**	**0.013 ± 0.014**	0.143 ± 1.18	3.295 ± 6.422	**94.077 ± 17.432**
Binary unit	15.062 ± 15.768	86.17 ± 116.14	0.118 ± 0.676	0.199 ± 0.392	**0.133 ± 1.15**	**3.289 ± 6.416**	93.926 ± 17.627

Table 12: Continuous versus binary contact weights, fidelity and correction magnitude. Fidelity columns follow Table 1(a) on the same 662 paired motions. Solver success is reported over all 663 attempted motions, and the binary variant leaves one motion with an infeasible coupled QP. Correction RMS is measured against the nominal trajectory the projector corrects.

	Kinematic fidelity					Correction RMS			Solver
Weighting	MPKPE (cm)↓	W-MPKPE (cm)↓	MPKVE (cm/s)↓	End (cm)↓	PCK@10 (%)↑	Root (cm)	Root (°)	Joint (°)	success
Continuous (ours)	**7.015 ± 0.501**	**13.156 ± 12.712**	**11.879 ± 12.595**	**4.701 ± 0.970**	**84.749 ± 4.424**	0.833	1.481	0.404	**663/663**
Binary unit	7.214 ± 0.586	13.562 ± 13.349	13.776 ± 13.484	5.182 ± 1.224	83.724 ± 4.921	1.198	2.159	0.736	662/663

E.3 Objective Terms

Table 13 removes one term or constraint family of Equation 3 at a time and keeps the nominal motion, the predicted modes, the compiled events, the remaining terms, and the solver fixed. Each

variant is run on the full test set under the protocol of Table 1. Joint limits stay exact in every variant, since the box projection of Appendix B.4 enforces them independently of the objective.

Each term prevents the artifact it targets, and no other term compensates for it. Without the tangential stick term, support-phase sliding rises to 32.83%, above the 20.60% of the nominal trajectory itself. The other corrections therefore move the feet along the ground, and only the stick term holds them in place. Without the ground constraint, penetration returns on more than a third of the contact samples.

The normal contact term trades a small amount of fidelity for contact. Without it, sliding and MPKPE decrease slightly, but floating increases and whole-body contact preservation falls by almost five points. This is the role of the two-sided residual of Equation 5, which pulls the active patch onto the surface instead of only keeping it above it.

Table 13: Leave-one-out ablation of the CAP objective on the test set. Each row removes one term or constraint family of Equation 3 and keeps the nominal motion, the predicted modes, the compiled events, the remaining terms, and the solver fixed. Values follow the protocol of Table 1(b), as mean $\pm$ standard deviation over sequences. Every variant has zero joint-limit violation.

	Foot skating		Penetration		Floating		Contact pres.
Variant	**Dur. (%)↓**	**Max (cm/s)↓**	**Dur. (%)↓**	**Max (cm)↓**	**Dur. (%)↓**	**Max (cm)↓**	**(%)↑**
Full CAP	4.39 ± 5.83	77.84 ± 83.14	0.00 ± 0.00	0.01 ± 0.01	0.14 ± 1.18	3.29 ± 6.42	94.09 ± 17.42
w/o tangential stick	32.83 ± 20.88	89.91 ± 72.83	0.00 ± 0.00	0.01 ± 0.01	0.14 ± 1.18	3.13 ± 6.45	94.22 ± 17.26
w/o normal contact	3.85 ± 5.63	80.08 ± 85.80	0.00 ± 0.00	0.00 ± 0.01	0.24 ± 1.47	4.91 ± 6.27	89.44 ± 20.88
w/o ground constraint	4.24 ± 5.76	76.87 ± 81.48	35.34 ± 20.09	2.31 ± 1.17	0.14 ± 1.17	2.91 ± 6.44	94.46 ± 17.10

E.4 Number of Gauss–Newton Sweeps

Table 14 varies the number of Gauss–Newton (GN) sweeps on the test set (239,888 frames at 30 Hz). Let $\mathcal{Q}_n$ be the trajectory after n sweeps and $\mathcal{Q}_{10}$ the 10-sweep reference. We report the normalized solution-residual gap

$$g_n = \frac{\|\mathcal{Q}_n - \mathcal{Q}_{10}\|_{\mathbf{W}_q}}{\|\mathcal{Q}_0 - \mathcal{Q}_{10}\|_{\mathbf{W}_q}}, \tag{22}$$

where the norm compares trajectories in the 35-dimensional tangent space, pooled over all frames and weighted by the proximal tolerances of $\mathbf{W}_q$ (4 cm for root translation, 0.15 rad for root rotation and joint angles). The relative step uses the same denominator and the difference from the preceding reported budget (0, 1, 2, 3, 5, or 10 sweeps), so the last two steps span two and five sweeps. The 10-sweep output is an empirical reference, not a solution certified by a stopping tolerance.

Contact quality converges after a single sweep. One sweep already removes all pooled whole-body penetration and brings sliding to the level of the full solve, and neither metric changes for any larger budget. Additional sweeps only reduce the distance to the 10-sweep reference. The residual gap falls from about 8% after one sweep to under 1.5% after three, and later sweeps contribute little. We use three sweeps because this leaves a clear convergence margin at about half the cost of the 10-sweep solve. A single sweep would halve the optimizer time again without changing the contact metrics. The 4.5 ms of Table 2(b) is the same three-sweep stage measured on the LAFAN1 sequences, and the three-sweep sliding value rounds to the 4.39% of Table 1(b). Note that this study pools contact samples across sequences whereas Table 1(b) reports sequence-wise means, so the two penetration columns differ in aggregation rather than in criterion.

Table 14: Truncated GN solves on the test set. Sliding is the share of source-contact foot samples with tangential speed above 10 cm/s, and penetration duration is the share of source-contact group–frame samples in which the group reaches more than 1 cm below the ground plane, over all twelve body groups. Optimizer-only time covers the 32-frame windows with 16-frame overlap and output assembly, and excludes the nominal retargeter, contact inference and compilation, and file I/O. Timings used one sequential Python worker on an Intel Core Ultra 9 285K CPU (24 cores), with CPU affinity and BLAS thread limits not fixed.

n_{GN}	Residual gap (%)↓	Rel. step (%)↓	Sliding (%)↓	Pen. dur. (%)↓	Time (ms/frame)↓
0	100.000	–	20.603	50.398	0.000
1	7.917	101.259	4.419	0.000	2.520
2	2.579	8.250	4.378	0.000	3.404
3	**1.426**	**2.456**	**4.387**	**0.000**	**4.881**
5	1.063	1.308	4.390	0.000	6.586
10	0.000	1.063	4.386	0.000	10.153