

Towards Contact-Rich Humanoid Dance: Bringing the Schuhplattler to the Unitree G1

Hannes Vogel
Autonomous Vehicle Systems Lab
Technical University of Munich
Munich, Germany
hannes.vogel@tum.de

Finn Rasmus Schäfer
Autonomous Vehicle Systems Lab
Technical University of Munich
Munich, Germany
finn.schaefer@tum.de

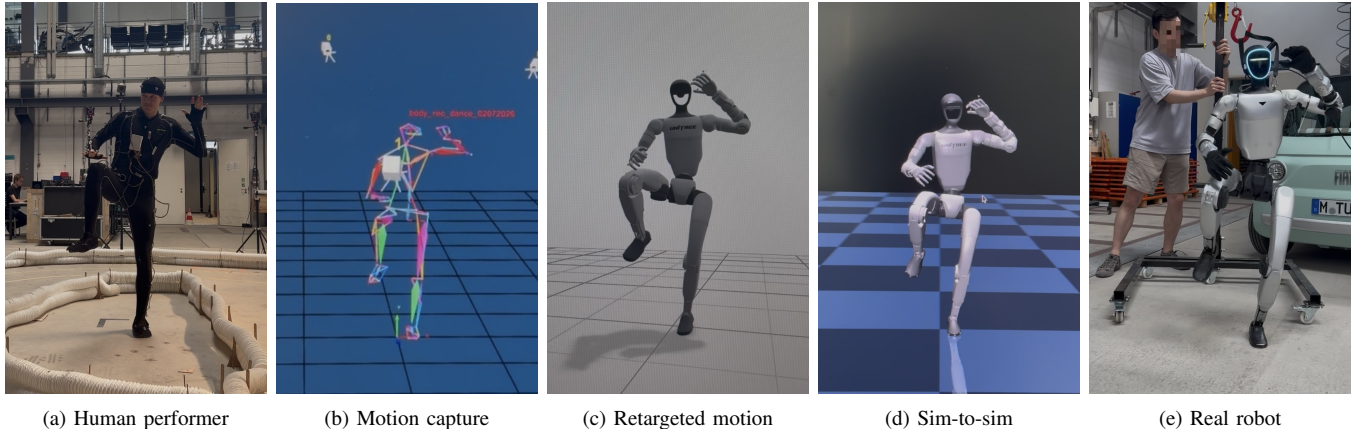


Fig. 1. From human performance to robot deployment: (a) the Schuhplattler performed by a human, (b) recorded with an optical motion capture system, (c) retargeted to the Unitree G1, (d) executed by the trained tracking policy in MuJoCo, and (e) on the real robot.

Abstract—Humanoid robots can now reproduce highly dynamic human motions such as dances and martial arts, yet motions involving intentional contact with the robot’s own body remain largely unaddressed, as standard motion-tracking pipelines typically treat self-contact as a condition to be avoided. We study this gap using the Bavarian Schuhplattler, a folk dance characterized by rhythmic slaps of the hands against the thighs and shoes. We adapt the open-source Holosoma framework to process custom optical motion capture recordings and correct a coordinate-conversion bug that mirrored all retargeted motions. We then deploy whole-body tracking policies for a Schuhplattler sequence on the 29-degree-of-freedom Unitree G1, secured by a safety gantry. Enabling self-collision in training has no measurable effect on tracking performance, but standard tracking rewards provide no incentive for the slaps. We further identify where wrist rotation, which is essential for the slaps, is lost in the pipeline: in the position-only reference used for training, the wrist orientation deviates from the recorded motion by 30° for the left and 86° for the right wrist on average, and a prototype orientation extension only partially resolves this issue. Based on these findings, we propose a contact-aware tracking reward as a next step towards intentional self-contact in humanoid dance.

I. INTRODUCTION

Humanoid robots trained with reinforcement learning to imitate human reference motions now perform highly dynamic skills such as dances, martial arts, and acrobatics on real hardware, for instance on the Unitree G1 [1]–[3]. These motions, however, are largely free of deliberate physical contact beyond

the feet touching the ground. Motions that involve contact-rich interaction remain considerably more challenging, since success then depends not only on reproducing the reference kinematics but also on establishing contact at the right moment, in the right place, and with appropriate force [4].

A particularly underexplored case of contact-rich motion is *intentional self-contact*, where the robot deliberately strikes its own body as part of the motion. To our knowledge, this case has not yet been addressed explicitly in humanoid motion tracking. Instead, standard motion tracking pipelines treat self-contact as something to be avoided: in the default configuration of the KungfuBot pipeline [2], contacts on the arm links are penalized and contacts on the upper legs terminate the training episode, and Holosoma [5] likewise penalizes self-contact as an undesired contact. This raises the question: *Can humanoid whole-body controllers learn and execute dynamic motions involving intentional self-contact while maintaining balance and physical consistency?* In search of a suitable testbed for this question, the choice at a Bavarian university was an obvious one: the traditional *Schuhplattler* folk dance. Beyond its cultural appeal, the dance is technically demanding. Fast, rhythmic slaps of the hands against the thighs and shoes, pronounced wrist rotations, and repeated single-leg phases stress balance, tracking precision, and self-contact at the same time.

In this work, we take a first step towards answering this

research question. Starting from the literature on humanoid motion tracking, we assembled a pipeline from existing open-source components to bring the Schuhplattler onto the 29-degree-of-freedom (DoF) Unitree G1. Retargeting our own motion capture recordings to the robot proved to be the main bottleneck, which none of the retargeting approaches we evaluated resolved satisfactorily, until we adopted the Holosoma framework [5].

This report makes three contributions. First, we adapt Holosoma to our motion capture recordings, including a fix for a coordinate-conversion bug that mirrored all retargeted motions, and deploy the resulting policies for a short dance sequence on the real G1 (Fig. 1). Second, we trace where wrist rotation, essential for the slaps, is lost along the pipeline and test a prototype extension that restores it only partially. Third, we compare training with and without self-collision, analyze why standard tracking does not produce the slaps, and propose a contact-aware reward.

This work is thus a *proof of concept*: it shows that the Schuhplattler can be brought onto a real humanoid with existing tools, while wrist rotation and intentional contact remain open. It provides the baseline on which we plan to build towards answering the research question posed above.

II. RELATED WORK

A. Physics-Based Motion Imitation

Learning control policies by imitating human reference motions with reinforcement learning was popularized by DeepMimic [6], which combines a pose-tracking reward with reference-state initialization and early termination to reproduce motion capture clips in simulation, and by concurrent work on physics-based motion capture imitation [7]. PHC [8] scaled this approach to large motion datasets for simulated humanoid avatars. An alternative line of work replaces explicit tracking rewards with adversarial style rewards learned from motion data [9], [10]. We follow the tracking-reward formulation, which underlies the real-robot systems discussed below.

B. Humanoid Motion Tracking on Real Hardware

Recent work transfers motion tracking to real humanoid robots, most prominently the Unitree G1. KungfuBot [2] learns highly dynamic skills such as martial arts and dance from video-extracted reference motions, and KungfuBot2 [11] extends it to longer and more diverse motion sequences. BeyondMimic [1] proposes a compact tracking reward that transfers to dance and acrobatic motions without per-motion tuning, and RobotDancing [3] uses residual actions, building on earlier residual formulations [12], for robust long-horizon dance tracking. ExBody2 [13] and SONIC [14] target expressive and large-scale whole-body tracking, and further works pursue universal motion trackers [15]–[17] or whole-body teleoperation [18], [19]. ASAP [20] addresses the sim-to-real gap by learning a model of the real robot’s dynamics mismatch. Our work builds on Holosoma [5], an open-source framework that integrates retargeting, policy training, sim-to-sim validation, and real-robot deployment for the G1, and

whose whole-body tracking configuration follows BeyondMimic.

C. Motion Acquisition and Retargeting

Reference motions are obtained either from video or from motion capture. Video-based pipelines reconstruct human motion with methods such as GVHMR [21], which KungfuBot uses as its motion source, but fine details such as wrist rotation are difficult to recover from monocular video. The reconstructed motion must then be retargeted to the robot’s morphology. GMR [22] shows that retargeting quality strongly affects downstream tracking success, learning-based methods such as IKMR [23] enable fast retargeting at scale, and ReActor [24] incorporates physics into retargeting to eliminate artifacts such as foot sliding and self-penetration. Contact can also be preserved explicitly within the retargeting objective, as in the foot-contact loss of WBMI [25]. For wrist-intensive motions, two distinct limitations arise. First, some G1 pipelines, such as KungfuBot [2], operate in a 23-DoF configuration with locked wrists, so wrist rotation is not possible. Second, even with actuated wrists, retargeting objectives that primarily match keypoint positions or weakly weight orientation relative to position leave wrist rotation largely unconstrained, so it is lost before policy training begins, as we show in Section IV-D2.

D. Contact in Humanoid Motion Tracking

Beyond foot–ground contact, contact-rich humanoid behavior has primarily been studied in interactions with external objects. ResMimic [4] adds a residual policy on top of a pretrained general motion tracker and uses a contact reward derived from the reference motion to achieve whole-body loco-manipulation, and Imagine2Real [26] similarly trains a residual adaptor for contact precision on top of a frozen tracker. Self-contact, in contrast, is typically treated as a failure case: it is penalized as an undesired contact during training [2], [5], enforced as a collision-avoidance constraint at runtime [27], or removed from the reference during retargeting [24]. Even the contact-tracking reward of InterMimic avoids promoting self-contact, and self-collision is disabled entirely when transferring to humanoid robots [28]. Intentional self-contact has been studied in other settings: on humanoid robots, self-touch serves kinematic calibration [29] and body-model learning [30] through quasi-static reaching, and contact-aware retargeting preserves self-contacts for animated characters, but purely kinematically [31]. To our knowledge, no prior work treats intentional self-contact as an objective of humanoid motion tracking, which is the gap this work begins to address.

III. PROBLEM STATEMENT AND SCOPE

Answering the research question posed in Section I for the Schuhplattler requires solving three sub-problems that build on each other:

- 1) **A faithful reference motion.** The dance must be captured and retargeted such that the details relevant for contact are preserved, in particular the wrist rotations

that orient the hand during a slap and the timing and location of the contacts.

- 2) **Stable tracking on real hardware.** A policy must track this reference on the real robot despite fast arm motion and repeated single-leg phases, without falling.
- 3) **Intentional contact.** The policy must actually produce the slaps, which standard tracking objectives do not require.

This report addresses the first sub-problem partially, since wrist rotation is not yet preserved, and the second fully, and analyzes the third.

IV. SYSTEM OVERVIEW

This section describes the pipeline, its hardware and data, and each of its stages, and concludes with the design decisions and alternatives that led to it.

A. Pipeline

Teaching the robot the Schuhplattler through motion imitation requires a pipeline that covers five stages: (i) recording a human reference motion, (ii) retargeting it to the kinematics of the G1, (iii) training a tracking policy in simulation, (iv) validating the policy in a second simulator, and (v) deploying it on the real robot. The reference motion can in principle be obtained from video or from motion capture. We use optical motion capture for its accuracy, in particular for wrist rotations and foot contacts (Section IV-H).

Since stages (ii)–(v) are well studied, we build on HoloSoma [5], an open-source framework that covers all four within a single codebase and required only minor adaptations for our use case (Section IV-D). The resulting pipeline is shown in Fig. 2. Motion capture recordings are exported as BVH files and retargeted to G1 joint trajectories. A whole-body tracking policy is then trained with Proximal Policy Optimization (PPO) [32] in IsaacSim, validated in MuJoCo [33] with the robot initially suspended on a virtual gantry, and finally deployed on the real G1 using the same setup.

B. Hardware

We use the Unitree G1 EDU humanoid robot [34] (Fig. 1e), which is 1.32 m tall and weighs approximately 35 kg. In our configuration, it has 29 actuated degrees of freedom: six per leg, three in the waist, and seven per arm, of which three form the wrist (roll, pitch, and yaw). Unlike the 23-DoF variant with locked wrists used by KungfuBot [2], this configuration allows the hand rotations of the Schuhplattler. The policy outputs target joint positions at 50 Hz, which are tracked by joint-level PD controllers on the robot. For both sim-to-sim validation and the hardware trials, the robot is suspended from a gantry that catches it in the event of a fall, and the simulated setup mirrors this with a virtual gantry.

C. Motion Capture Data

Reference motions were recorded with an optical motion capture system from NOKOV [35], using eight cameras at 60 Hz and the XINGYING 3.1 software, which tracks reflective markers on the performer and fits a skeleton to them.

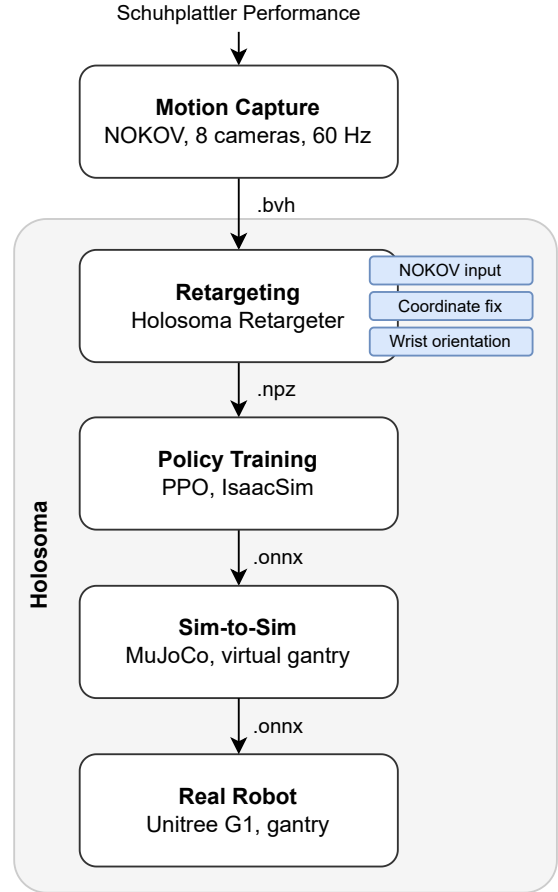


Fig. 2. Overview of the pipeline, built on the HoloSoma framework [5]. Our additions are shown in blue. The wrist orientation extension is a prototype and was not used for policy training.

We use NOKOV’s 53-marker full-body template, which fits a 23-segment skeleton (Figs. 1a and 1b). Two markers on each wrist and two on each hand define the orientation of the hand segment, so that the wrist rotations of the Schuhplattler are captured explicitly, in contrast to the video-based extraction discussed in Section IV-H.

The first author recorded 22 takes, covering full Schuhplattler sequences as well as isolated elements such as slaps, leg changes, and wrist turns. Each take is exported as a BVH file with the skeleton hierarchy and per-frame joint rotations. Its joint naming largely follows the convention of the LAFAN1 dataset [36], which allowed us to reuse HoloSoma’s existing LAFAN-to-G1 joint mapping (Section IV-D).

D. Retargeting

Retargeting maps the recorded human motion onto the kinematics of the G1. HoloSoma’s retargeter [5] takes world-frame joint positions of the human skeleton and solves, frame by frame, for a robot configuration whose body links match them, subject to the joint limits and a foot-sticking constraint that suppresses foot sliding. We added an input option for

our NOKOV recordings, which reuses the existing LAFAN-to-G1 joint mapping and BVH forward-kinematics routine (Section IV-C). The recordings are downsampled from 60 Hz to 30 Hz, and the motion is scaled uniformly by the ratio of the robot’s height to the performer’s height (1.92 m). All 22 takes were retargeted in this way and converted to the 50 Hz format used for policy training.

1) *Coordinate-Conversion Bug*: When retargeting our first takes, every motion appeared mirrored front-to-back, with the legs flipped by 180° and the arms raised backward. The cause was the retargeter’s conversion from the Y-up frame of the motion capture data to the robot’s Z-up frame, which used a reflection (determinant -1) instead of a rotation. Replacing it with a rotation of 90° about the x -axis resolved the issue, also for Holosoma’s existing LAFAN input, which shares the same conversion.

2) *Wrist Orientation Tracking*: The Schuhplattler relies on wrist rotations that orient the palm towards the thigh or shoe during a slap. We found that wrist rotation can be lost at three stages of the pipeline.

Motion extraction from video. In the video-based pipeline evaluated first (Section IV-H), GVHMR [21] estimates wrist rotation from body keypoints only. In a test recording with more than 90° of wrist rotation, it recovered less than 10°, and a whole-body keypoint model recovered the rotation only partially and with noticeable jitter. This motivated the switch to optical motion capture.

Weighted orientation costs. In the IK-based retargeter of the same pipeline, the wrist orientation cost was about $300\times$ weaker than the position costs, and a posture term pulled the wrists back towards zero, so that less than 5% of the source rotation was retained. We therefore injected the human wrist rotation, taken relative to the elbow, into the robot’s three wrist joints after the IK solve.

Position-only retargeting. Holosoma’s retargeter matches positions only and discards wrist rotation entirely. We prototyped an orientation-tracking term for the hands, restricted to the wrist joints by masking its Jacobian. This required removing hard-coded wrist joint limits, fixing an initialization error, and handling a kinematic singularity: since the G1 wrist joints act in series as roll, pitch, and yaw, the roll and yaw axes align as the pitch joint approaches its limit near 90°, which destabilizes the solver during pronounced rotations. The extension is evaluated in Section V-A and was not used for policy training.

E. Policy Training

We train the tracking policy with Holosoma’s whole-body tracking configuration, which uses PPO in IsaacSim. The result is a single neural network policy that controls all 29 joints of the robot. At each control step, it receives the robot’s proprioceptive state together with the current target frame of the reference motion, and outputs target positions for all joints, which are tracked by the PD controllers described in Section IV-B. The reward follows the structure of Beyond-Mimic [1]: it combines terms that track the reference body

positions, orientations, and velocities with penalties on abrupt action changes, joint-limit violations, and undesired contacts. To improve robustness for transfer to hardware, training uses random pushes and randomized friction and center-of-mass offsets. The policy is motion-specific: it is trained on a single reference clip rather than as a general-purpose motion tracker. Each policy is trained with 4096 parallel environments for up to 30,000 iterations, which takes about 15 h on a single NVIDIA RTX 5090.

We trained policies on one take, both on the full clip and on a trimmed version without the initial calibration pose, which is instead extended by interpolated transitions from and to the robot’s default pose. Each variant was trained with and without self-collision. With self-collision enabled, the simulator treats the robot’s links as solid bodies, so the hands are blocked by the thighs. Without it, the links pass through each other, and only contact with the ground is simulated. The resulting policies are compared in Section V-B. After training, all policies are exported to ONNX format for deployment.

F. Sim-to-Sim Validation

Before deployment on hardware, each policy is evaluated in MuJoCo, whose contact and actuator models differ from those of IsaacSim, as an inexpensive check for overfitting to the training simulator. Holosoma runs the policy through the same inference pipeline and robot interface as on the real G1, so the procedure mirrors the hardware deployment. The procedure therefore mirrors the hardware deployment. The simulated robot starts suspended from a virtual gantry, which is lowered until the feet bear weight and then released. The policy is then activated, the motion clip is started, and once the clip ends, control is handed over to a fallback locomotion policy that keeps the robot standing, since the tracking policy is only defined for the duration of the clip.

G. Real-Robot Deployment

For hardware deployment, the exported ONNX policy runs offboard on a laptop connected to the robot via Ethernet, with the PD gains loaded from the model’s metadata so that they match those used during training. The operator uses a joystick to activate the policy, start the motion clip, switch to the fallback locomotion policy, and trigger an emergency stop.

We followed a staged validation protocol. A locomotion policy provided with Holosoma was run first to validate the robot, network, and inference setup independently of our own policies. Our Schuhplattler policies, both trained with self-collision enabled, were then deployed with the robot suspended from the gantry throughout. Before each run, we verified that the robot’s feet carried its full weight (Section V-C).

H. Design Decisions and Evaluated Alternatives

The system described above is the result of an iterative process: starting from the literature, we adopted an existing open-source pipeline, adapted it to the Schuhplattler, and replaced components whenever a stage proved to be a blocker.

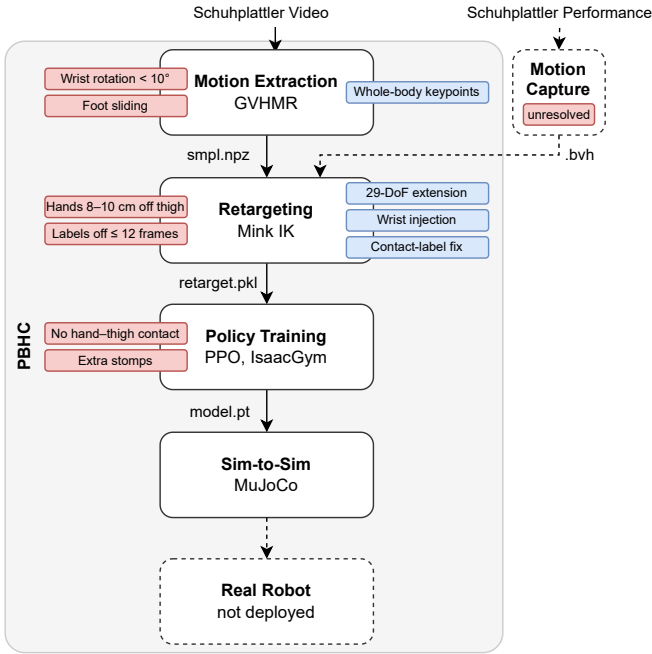


Fig. 3. Video-based pipeline evaluated first, built on KungfuBot [2]. Our additions are shown in blue, and problems identified in our stage-by-stage analysis in red. Motion capture data could not be integrated into this pipeline (dashed), and it was not deployed on hardware.

This subsection summarizes the alternatives we evaluated and why they were abandoned.

1) *Video-Based Pipeline*: We initially built on the open-source pipeline of KungfuBot [2], which covers the same stages as our final system but uses monocular video as its motion source (Fig. 3). Human motion is reconstructed from video with GVHMR [21], retargeted to the robot by inverse kinematics, and used to train a tracking policy with PPO in IsaacGym [37], followed by sim-to-sim validation in MuJoCo. The pipeline targets a 23-DoF configuration of the G1 with locked wrists. Since wrist rotation is essential for the Schuhplattler, we extended the robot model and the retargeting to the 29-DoF robot, with additional hand targets, a reworked keypoint mapping, and the wrist-injection method described in Section IV-D2. A corresponding 29-DoF training configuration was prepared, but not used for training before we switched pipelines.

To identify the weaknesses of this pipeline, we analyzed trained policies and traced their failures back through each stage, from the policy to the retargeted motion, the extracted human motion, and the input video. This revealed problems at every stage. The extracted motion showed foot sliding after the feet touched down and unreliable wrist rotation (Section IV-D2). The retargeting removed the hand-thigh contacts present in the extracted motion: at the slap frames, both hands remained 8 cm to 10 cm from the thigh surface. The automatically detected foot-contact labels were off by up to 12 frames at transitions and systematically wrong at the start and end of the clips, which we corrected with a post-

TABLE I
POLICIES TRAINED WITH THE VIDEO-BASED PIPELINE (23-DOF G1, LOCKED WRISTS). FINAL VALUES AT THE END OF TRAINING, WITH THE BEST IN BOLD AND THE SECOND BEST UNDERLINED. THE SELF-CONTACT PENALTY IS NOT RANKED, AS FEWER SELF-CONTACTS ARE NOT DESIRABLE FOR THE SCHUHPLATTLER.

	Single slap 50k it.	Sequence 120k it.	Sequence 171k it.
Motion completion	0.84	0.98	0.99
Hand tracking error [cm]	9.0	<u>5.7</u>	5.2
Joint error [rad]	1.15	<u>0.91</u>	0.86
Self-contact penalty	-1.70	-0.06	-0.04

processing script.

Table I summarizes three policies trained on this data. A policy trained on a short single-slap sequence frequently lost balance at the slap following a leg lift, whereas policies trained on a longer sequence of about 9 s completed nearly the entire motion with almost no falls. None of them produced hand-thigh contact, and for the two sequence policies, the self-contact penalty remained close to zero. The better-performing policies also inserted stomps absent from the reference, which the foot-contact reward did not penalize, as it does not distinguish reference contacts from spurious ones. Since several of these problems originate in the video-based motion source itself, we decided to record the dance with optical motion capture, which measures wrist rotations and foot contacts directly.

2) *Retargeting Motion Capture Data*: Integrating motion capture data proved to be the main bottleneck. The video pipeline expects SMPL-based human motion, whereas our recordings are skeletons in BVH format. Our own converter, which uses the BVH joint positions directly as inverse-kinematics targets and injects the wrist rotations from the BVH joint rotations, transferred wrist rotation on a test take, but with a mean keypoint error of 24 cm (15 cm after aligning the pelvis) and foot-contact labels that marked both feet as grounded throughout. Further approaches, namely the general-purpose retargeter GMR [22], conversion of the BVH skeleton to SMPL, and fitting SMPL bodies to the marker data in the style of AMASS/MoSh++, did not yield satisfactory results for our data either. We therefore switched to Holosoma [5], which provides the same pipeline stages as the KungfuBot pipeline, supports the 29-DoF G1 natively, and, crucially, includes a retargeter with a tested entry point for BVH motion capture data. Only the recorded motion capture data was carried over from the previous pipeline.

V. EXPERIMENTS AND RESULTS

We evaluate the retargeted reference motion, policy training and sim-to-sim transfer, and the deployment on the real robot. Quantitative results come from the retargeted motions and the training logs, while the behavior in simulation and on hardware is assessed qualitatively.

A. Retargeting Results

All 22 motion capture takes were retargeted to the G1. The policies were trained on a complete Schuhplattler take

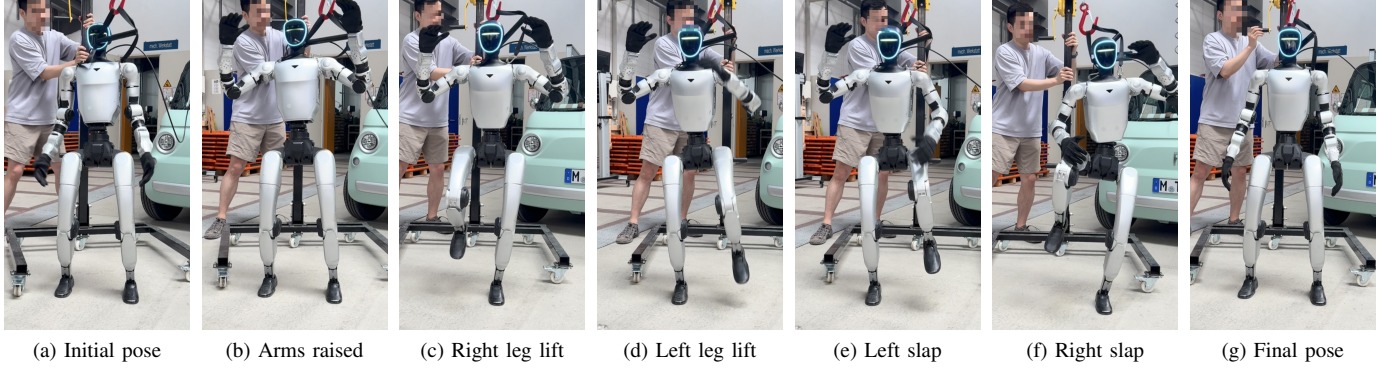


Fig. 4. Deployment of a trained tracking policy (self-collision enabled) on the real Unitree G1, with the gantry as a safety measure. The robot raises its arms (b), lifts its right leg for a stomp (c), lifts its left leg (d), slaps towards the left thigh (e), repeats the slap on the right side (f), and returns to the final pose (g). Left and right refer to the robot’s sides.

retargeted with the position-only objective. In this reference, the wrist joints move by only 6° to 23° over the entire take, while the performer’s wrists rotate by up to 175° . Over the trained segment, the mean wrist orientation error with respect to the motion capture data is 30° for the left and 86° for the right wrist, so wrist rotation is largely absent from the reference the policies learn to track.

The prototype orientation extension (Section IV-D2) reduces the left-wrist error to 0.2° , but the right-wrist error remains between 57° and 127° . Moreover, a low orientation error does not imply a natural pose: some frames reach the recorded hand orientation only through anatomically implausible wrist configurations, and the extension also changes the shoulder and elbow angles by up to 29° , as the position terms compensate for the rotated hand. Preserving wrist rotation in a plausible way therefore remains an open problem.

B. Policy Training and Simulation Results

Table II summarizes the four policies trained with HoloSoma. All of them converged to full-length episodes, i.e., the policies track the entire reference without early termination. Within each pair, enabling self-collision had no measurable effect on the training outcome: the episode return differs by less than 1%, and the tracking errors differ by at most 1 mm in body position and 0.011 rad in joint angles. Training on the trimmed clip with transitions reduced the body position error from 3.3 cm to 2.9 cm and the joint error from 0.50 rad to 0.45 rad. While the body poses relative to the torso are tracked closely, the global torso position deviates by about 16 cm, making it the least accurately tracked quantity.

Fig. 5 shows the training curves of the second pair. Without self-collision, the undesired-contact penalty remains zero throughout, since the only possible contact is with the ground, which the penalized bodies do not touch. With self-collision, the number of penalized bodies in contact peaks early in training, while the episodes grow from a few dozen to the full 500 steps, and then settles at a low level, corresponding to about 0.01 to 0.015 penalized bodies in contact per control step. Since the thigh link is among the penalized bodies,

TABLE II
FINAL TRAINING RESULTS OF THE HOLOSOMA POLICIES (MEAN OVER THE LAST 100 ITERATIONS). POLICIES 1A/1B USE THE FULL CLIP, 2A/2B THE TRIMMED CLIP WITH TRANSITIONS. WITHIN EACH PAIR, ONLY SELF-COLLISION DIFFERS.

Policy	1a	1b	2a	2b
Self-collision	off	on	off	on
Episode return	41.2	41.0	41.9	41.5
Episode length [steps]	500	500	500	499
Body position error [cm]	3.3	3.4	2.9	2.9
Torso position error [cm]	16.6	16.1	16.3	16.3
Body orientation error [rad]	0.114	0.116	0.101	0.103
Joint error [rad]	0.50	0.51	0.45	0.46
Undesired contacts [1/s]	0	0.52	0	0.72

hand–thigh contacts contribute to this penalty, so the reward provides no incentive to produce them. With a weight of -0.1 , however, the penalty accounts for only about 0.03% of the episode return. Any differences in behavior between the policies with and without self-collision therefore stem mainly from the simulated physics, in which the arms are blocked by the body, rather than from the penalty.

C. Real-Robot Deployment

After passing the sim-to-sim check in MuJoCo, both policies trained with self-collision, 1b on the full clip and 2b on the trimmed clip, were deployed on the real G1 with the gantry as a safety measure and executed the trained sequence (Fig. 4). The motion is clearly recognizable as the Schuhplattler, including the leg lifts, the stomp, and the arm swings towards the thighs, but without wrist rotation, since the policies were trained on the position-only reference. Contact between hands and thighs was not measured.

The main practical issue concerned the gantry itself. When the robot was suspended such that its feet touched the ground but most of its weight was carried by the gantry, the policy twitched and did not track the motion properly. After lowering the robot until its feet carried its full weight, the policy ran without issues.

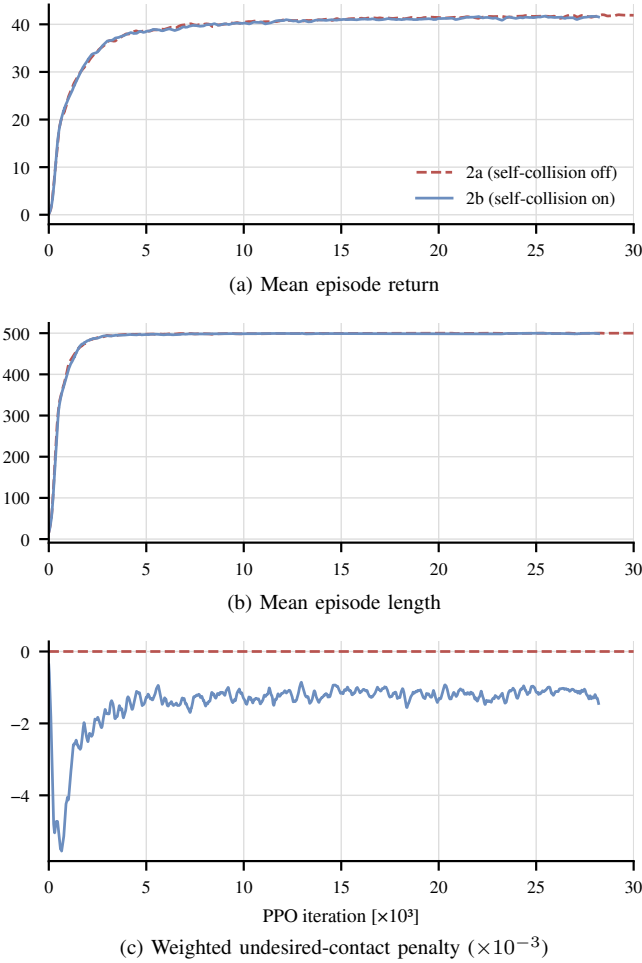


Fig. 5. Training curves of the policies trained on the trimmed clip without (2a) and with (2b) self-collision, smoothed with a 200-iteration moving average. Policy 2b was stopped at 28,000 iterations.

VI. TOWARDS INTENTIONAL SELF-CONTACT

Our results show that a standard tracking pipeline reproduces the kinematics of the Schuhplattler on real hardware, but nothing in it rewards the dance’s defining feature, the slaps. This section analyzes why and outlines next steps.

A. Why Standard Tracking Does Not Produce the Slaps

We identified six obstacles, each of which alone can prevent intentional self-contact:

- **No incentive:** the reward only favors matching the reference, and hand–thigh contact is even weakly penalized (Section V-B).
- **Infeasible reference:** kinematic retargeting does not enforce contact. In the video-based pipeline, the hands stopped 8 cm to 10 cm short of the thighs (Section IV-H).
- **Missing wrist rotation:** the wrist orientation in our reference deviates from the recording by 30° and 86° on average (Section V-A).
- **Rare contact windows:** each slap lasts only a fraction of a second.

- **Coarse contact signals:** contact terms count bodies above a force threshold, e.g., 1 N in Holosoma, and cannot distinguish a brush from a slap.
- **Contact-based termination:** some pipelines end the episode on upper-leg contact [2].

B. A Contact-Aware Tracking Reward

Addressing the first, fourth, and fifth obstacles, the tracking reward could be extended by a contact term with weight w_c ,

$$r_t^{\text{contact}} = \sum_{i \in \mathcal{P}} \hat{c}_t^{(i)} \min\left(\frac{f_t^{(i)}}{f_{\text{ref}}}, 1\right), \quad (1)$$

where $\mathcal{P}$ is the set of intended contact pairs, such as the left hand and the left thigh, $f_t^{(i)}$ is the simulated contact force of pair i , and $\hat{c}_t^{(i)} \in \{0, 1\}$ marks the contact windows, derived by thresholding the distance between the two bodies in the reference. A related contact reward is used by ResMimic for object interaction [4]. Saturating at f_{ref} avoids rewarding ever harder strikes. In addition, w_c must be large enough to matter, given that the existing contact penalty accounts for only about 0.03 % of the return. The intended pairs must also be exempted from that penalty, and episode start times should be sampled near the contact windows. Since the hand is modeled as a single capsule, however, the simulated contact would represent the end of the forearm striking the thigh rather than a slap with the palm.

Beyond the reward, contact-aware retargeting could make the reference itself contact-consistent [24], [25], or a residual policy could add the contact on top of the frozen tracking policy [4]. In any case, contact must be verified explicitly, in simulation through the pairwise contact forces and on hardware, for instance, through synchronized video and audio, since a slap is audible.

VII. CONCLUSION

This work brought the Schuhplattler onto the real Unitree G1 and analyzed what still prevents motion tracking with intentional self-contact, the capability our research question asks for. Three lessons stand out. First, enabling the wrist joints is not sufficient if the motion source and retargeting do not preserve wrist rotation. Second, the main bottleneck was not policy training but obtaining a faithful reference motion. Third, seemingly minor details decide success, from a single reflected coordinate axis that corrupted every retargeted motion to a gantry that carried too much of the robot’s weight and made the policy twitch. The policies were trained on a single take and evaluated only with the gantry. Future work includes the proposed contact-aware reward, wrist retargeting under anatomical constraints, and training on all 22 takes towards a free-standing performance.

DECLARATION OF GENERATIVE AI AND AI-ASSISTED TECHNOLOGIES IN THE WRITING PROCESS

Declaration according to [38]: During the preparation of this work, the authors used Claude [39] to structure the report and to draft and revise text, Claude Code [40] to organize the

literature review, analyze the training logs, and generate the training-curve plots, and Grammarly [41] to check grammar. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the content of the paper.

REFERENCES

- [1] Q. Liao, T. E. Truong, X. Huang, Y. Gao, G. Tevet, K. Sreenath, and C. K. Liu, "BeyondMimic: From motion tracking to versatile humanoid control via guided diffusion," 2025, arXiv:2508.08241.
- [2] W. Xie, J. Han, J. Zheng, H. Li, X. Liu, J. Shi, W. Zhang, C. Bai, and X. Li, "KungfuBot: Physics-based humanoid whole-body control for learning highly-dynamic skills," 2025, arXiv:2506.12851.
- [3] Z. Sun, Y. Peng, Y. Meng, X. Li, B.-S. Huang, Z. Bing, X. Wang, and A. Knoll, "RobotDancing: Residual-action reinforcement learning enables robust long-horizon humanoid motion tracking," 2025, arXiv:2509.20717.
- [4] S. Zhao, Y. Ze, Y. Wang, C. K. Liu, P. Abbeel, G. Shi, and R. Duan, "ResMimic: From general motion tracking to humanoid whole-body loco-manipulation via residual learning," 2025, arXiv:2510.05070.
- [5] Y. Seo, C. Sferrazza, J. Chen, G. Shi, R. Duan, and P. Abbeel, "Learning sim-to-real humanoid locomotion in 15 minutes," 2025. [Online]. Available: <https://arxiv.org/abs/2512.01996>
- [6] X. B. Peng, P. Abbeel, S. Levine, and M. van de Panne, "DeepMimic: Example-guided deep reinforcement learning of physics-based character skills," *ACM Transactions on Graphics*, 2018.
- [7] N. Chentanez, M. Müller, M. Macklin, V. Makoviychuk, and S. Jeschke, "Physics-based motion capture imitation with deep reinforcement learning," in *Proceedings of the 11th Annual International Conference on Motion, Interaction, and Games*, ser. MIG '18. ACM, 2018.
- [8] Z. Luo, J. Cao, A. Winkler, K. Kitani, and W. Xu, "Perpetual humanoid control for real-time simulated avatars," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.
- [9] X. B. Peng, Z. Ma, P. Abbeel, S. Levine, and A. Kanazawa, "AMP: Adversarial motion priors for stylized physics-based character control," *ACM Transactions on Graphics*, 2021.
- [10] X. B. Peng, Y. Guo, L. Halper, S. Levine, and S. Fidler, "ASE: Large-scale reusable adversarial skill embeddings for physically simulated characters," *ACM Transactions on Graphics*, 2022.
- [11] J. Han, W. Xie, J. Zheng, J. Shi, W. Zhang, T. Xiao, and C. Bai, "KungfuBot2: Learning versatile motion skills for humanoid whole-body control," 2025, arXiv:2509.16638.
- [12] Y. Yan, E. Valls Mascaro, T. Egle, and D. Lee, "I-CTRL: Imitation to control humanoid robots through bounded residual reinforcement learning," *IEEE Robotics & Automation Magazine*, vol. 32, no. 1, pp. 59–67, 2025.
- [13] M. Ji, X. Peng, F. Liu, J. Li, G. Yang, X. Cheng, and X. Wang, "ExBody2: Advanced expressive humanoid whole-body control," 2025. [Online]. Available: <https://arxiv.org/abs/2412.13196>
- [14] Z. Luo, Y. Yuan, T. Wang, C. Li, F. Castañeda, S. Chen, Z.-A. Cao, J. Li, D. Minor, Q. Ben, J. Park, D. Sami, Z. Wang, X. Da, R. Ding, C. Hogg, L. Song, E. Lim, E. Jeong, T. He, H. Xue, W. Xiao, S. Yuen, J. Kautz, Y. Chang, U. Iqbal, L. Fan, and Y. Zhu, "SONIC: Supersizing motion tracking for natural humanoid whole-body control," 2025, arXiv:2511.07820.
- [15] K. Yin, W. Zeng, K. Fan, M. Dai, Z. Wang, Q. Zhang, Z. Tian, J. Wang, J. Pang, and W. Zhang, "UniTracker: Learning universal whole-body motion tracker for humanoid robots," 2025. [Online]. Available: <https://arxiv.org/abs/2507.07356>
- [16] Z. Lu, Z. Zheng, Y. Lyu, J. Liu, F. Zhang, S. Lu, X. Yuan, C. Sun, X. Zuo, and S. E. Li, "M3imic: Learning a versatile whole-body controller for multimodal motion mimicking," 2026. [Online]. Available: <https://arxiv.org/abs/2606.04829>
- [17] M. Chen, K. Wang, B. Zhang, X. Ma, Z. Yang, Y. Ren, Q. Huang, Z. Zhu, Y. Wang, and Z. Su, "HoloMotion-1 technical report," 2026, arXiv:2605.15336.
- [18] Y. Ze, Z. Chen, J. P. Araújo, Z.-a. Cao, X. B. Peng, J. Wu, and C. K. Liu, "TWIST: Teleoperated whole-body imitation system," 2025. [Online]. Available: <https://arxiv.org/abs/2505.02833>
- [19] Y. Ze, S. Zhao, W. Wang, A. Kanazawa, R. Duan, P. Abbeel, G. Shi, J. Wu, and C. K. Liu, "TWIST2: Scalable, portable, and holistic humanoid data collection system," 2025. [Online]. Available: <https://arxiv.org/abs/2511.02832>
- [20] T. He, J. Gao, W. Xiao, Y. Zhang, Z. Wang, J. Wang, Z. Luo, G. He, N. Sobanbab, C. Pan, Z. Yi, G. Qu, K. Kitani, J. Hodgins, L. Fan, Y. Zhu, C. Liu, and G. Shi, "ASAP: Aligning simulation and real-world physics for learning agile humanoid whole-body skills," 2025, arXiv:2502.01143.
- [21] Z. Shen, H. Pi, Y. Xia, Z. Cen, S. Peng, Z. Hu, H. Bao, R. Hu, and X. Zhou, "World-grounded human motion recovery via gravity-view coordinates," in *SIGGRAPH Asia 2024 Conference Papers*, 2024.
- [22] J. P. Araújo, Y. Ze, P. Xu, J. Wu, and C. K. Liu, "Retargeting matters: General motion retargeting for humanoid motion tracking," 2025. [Online]. Available: <https://arxiv.org/abs/2510.02252>
- [23] X. Chen, H. Wu, S. Wu, M. Zhou, D. Xiang, and H. Zhang, "A scalable whole-body motion transfer via implicit kinodynamic motion retargeting," 2025. [Online]. Available: <https://arxiv.org/abs/2509.15443>
- [24] D. Müller, A. Serifi, S. Christen, R. Grandia, E. Knoop, and M. Bächer, "ReActor: Reinforcement learning for physics-aware motion retargeting," *ACM Transactions on Graphics*, 2026.
- [25] Z. Chen, H. Zhang, D. Wang, J. Yu, H. Xu, Y. Wang, and R. Xiong, "A whole-body motion imitation framework from human data for full-size humanoid robot," 2025. [Online]. Available: <https://arxiv.org/abs/2508.00362>
- [26] J. Chen, Z. Wang, F. Jia, X. Chen, X. Niu, W. Zeng, T. Xue, X. Zhou, J. Pang, and J. Wang, "Imagine2Real: Towards zero-shot humanoid-object interaction via video generative priors," 2026. [Online]. Available: <https://arxiv.org/abs/2605.22272>
- [27] D. Morton, P. Mohnot, and M. Pavone, "Constrained whole-body tracking for humanoid robots," 2026, arXiv:2606.00374.
- [28] S. Xu, H. Y. Ling, Y.-X. Wang, and L.-Y. Gui, "InterMimic: Towards universal whole-body control for physics-based human-object interactions," 2025, arXiv:2502.20390.
- [29] A. Roncone, M. Hoffmann, U. Pattacini, and G. Metta, "Automatic kinematic chain calibration using artificial skin: Self-touch in the iCub humanoid robot," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 2305–2312.
- [30] F. Gama, M. Shcherban, M. Rolf, and M. Hoffmann, "Active exploration for body model learning through self-touch on a humanoid robot with artificial skin," 2020, arXiv:2008.13483.
- [31] R. Villegas, D. Ceylan, A. Hertzmann, J. Yang, and J. Saito, "Contact-aware retargeting of skinned motion," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 9720–9729.
- [32] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.
- [33] E. Todorov, T. Erez, and Y. Tassa, "MuJoCo: A physics engine for model-based control," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012, pp. 5026–5033.
- [34] Unitree Robotics. (2026) Unitree G1 Humanoid Robot. Accessed: Sep. 30, 2026. [Online]. Available: <https://www.unitree.com/g1>
- [35] NOKOV. (2026) XINGYING software documentation: 53 markers and skeleton descriptions. Accessed: Sep. 30, 2026. [Online]. Available: <https://xingying-docs.nokov.com/xingying/XingYing3.4-EN/-body-template/-53-markers-and-skeleton-descriptions/>
- [36] F. G. Harvey, M. Yurick, D. Nowrouzezahrai, and C. Pal, "Robust motion in-betweening," *ACM Transactions on Graphics*, vol. 39, no. 4, 2020.
- [37] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, and G. State, "Isaac Gym: High performance GPU-based physics simulation for robot learning," 2021, arXiv:2108.10470.
- [38] Elsevier. (2026) Artificial intelligence in the life sciences – guide for authors. Accessed: Sep. 30, 2026. [Online]. Available: <https://www.sciencedirect.com/journal/artificial-intelligence-in-the-life-sciences/publish/guide-for-authors>
- [39] Anthropic. (2026) Claude. Large language model (Claude Opus 5.5). Accessed: Sep. 29, 2026. [Online]. Available: <https://claude.ai>
- [40] ——. (2026) Claude Code. Agentic coding tool. Accessed: Sep. 29, 2026. [Online]. Available: <https://claude.com/product/claude-code>
- [41] Grammarly Inc. (2026) Grammarly. Accessed: Sep. 30, 2026. [Online]. Available: <https://app.grammarly.com/>