OmniRetarget: Interaction-Preserving Data Generation for Humanoid Whole-Body Loco-Manipulation and Scene Interaction

Lujie Yang*[1,2], Xiaoyu Huang*[1,3], Zhen Wu*[1],
Angjoo Kanazawa[†1,3], Pieter Abbeel[†1,3], Carmelo Sferrazza[†1], C. Karen Liu[†1,4], Rocky Duan[†1], Guanya Shi[†1,5]
[1]Amazon FAR (Frontier AI & Robotics), [2]MIT, [3]UC Berkeley, [4]Stanford University, [5]CMU

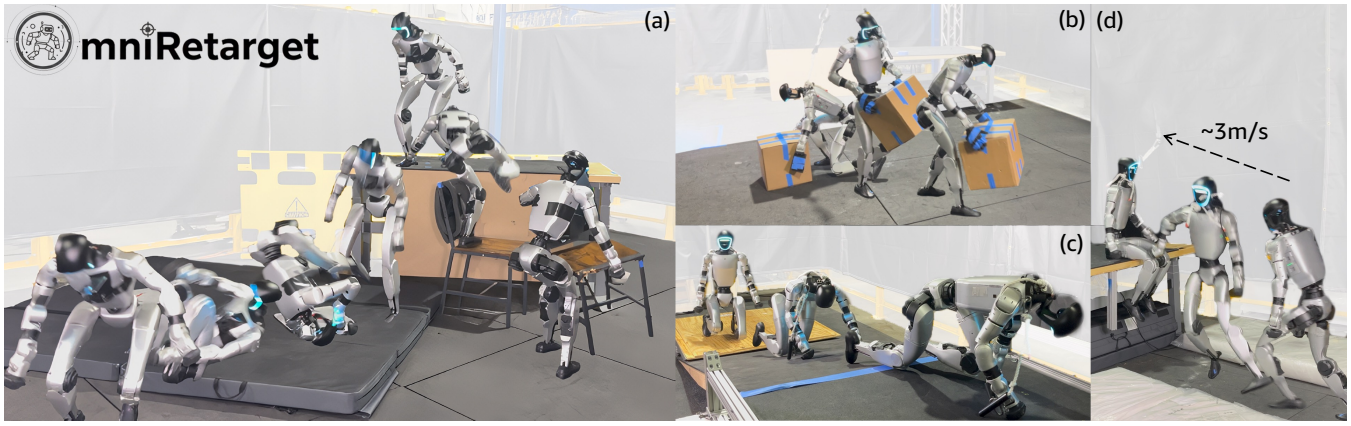

Fig. 1: OMNIRETARGET enables reinforcement learning policies to learn complex, long-horizon loco-manipulation skills in challenging environments that transfer zero-shot from simulation to a Unitree G1 humanoid. Thanks to the high-quality interaction-preserving motion retargeting, these policies are trained and deployed in a *minimal and unified* way: it involves only 5 rewards, 4 robot domain randomization terms, and a purely proprioceptive observation space, shared by all tasks. Demonstrated behaviors include **(a)** 30-second parkour course involving chair moving, stepping & vault, and jump & roll, **(b)** object transportation, **(c)** crawling on a slope, and **(d)** fast platform climbing and sitting.

***Abstract*— A dominant paradigm for teaching humanoid robots complex skills is to retarget human motions as kinematic references to train reinforcement learning (RL) policies. However, existing retargeting pipelines often struggle with the significant embodiment gap between humans and robots, producing physically implausible artifacts like foot-skating and penetration. More importantly, common retargeting methods neglect the rich human-object and human-environment interactions essential for expressive locomotion and loco-manipulation. To address this, we introduce OMNIRETARGET, an interaction-preserving data generation engine based on an interaction mesh that explicitly models and preserves the crucial spatial and contact relationships between an agent, the terrain, and manipulated objects. By minimizing the Laplacian deformation between the human and robot meshes while enforcing kinematic constraints, OMNIRETARGET generates kinematically feasible trajectories. Moreover, preserving task-relevant interactions enables efficient data augmentation, from a single demonstration to different robot embodiments, terrains, and object configurations. We comprehensively evaluate OMNIRETARGET by retargeting motions from OMOMO [1], LAFAN1 [2], and our in-house MoCap datasets, generating over 8-hour trajectories that achieve better kinematic constraint satisfaction and contact preservation than widely used baselines. Such high-quality data enables proprioceptive RL policies to successfully execute long-horizon (up to 30 seconds) parkour and loco-manipulation skills on a Unitree G1 humanoid, trained with only 5 reward terms and simple domain randomization shared by all tasks, without any learning curriculum. All code, retargeted datasets, and trained policies will be publicly released. Result videos can be found at https://omniretarget.github.io**

I. INTRODUCTION

The quest to enable humanoid robots to perform complex whole-body scene- and object-interaction tasks has long been constrained by a fundamental data bottleneck. While deep reinforcement learning (RL) has shown remarkable success in robot control, efficient exploration is highly sensitive to reward engineering [3]. This challenge is further amplified on humanoids, whose high-dimensional action spaces and complex dynamics make learning natural, expressive behaviors from scratch both difficult and inefficient.

To address these challenges, imitating human motions offers a powerful alternative for learning whole-body control, especially for complex scene interactions. Human demonstrations capture dynamic coordination, such as lifting objects while walking on uneven terrain, and have been used effectively in animation [4], [5], [6]. A critical challenge arises in robotics: unlike virtual characters, physical humanoids only approximate human morphology, with significant differences in shape, proportion and degrees of freedom. This embodiment gap means that simply adapting human motions is in-

* Equal contribution, work done while interning at Amazon FAR. † Amazon FAR team co-lead.

sufficient; it is essential to also adapt their scene interactions to the robot's specific form to generate usable references.

To this end, researchers have pursued two main strategies. The first one is teleoperation [7], [8], [9], where only a human operator's motions are retargeted to control the robot online. This approach leverages the human operator for real-time adaptation, which sidesteps the need for automatic interaction retargeting. However, despite the advantage of online feedback, the method remains labor-intensive and does not scale well for large-scale data generation. The second and more scalable strategy is offline interaction retargeting, which holistically adapts both the human's motion and their scene interactions to the robot's specific embodiment.

However, most existing retargeting methods [10], [9], [11] fall short in this regard. They predominantly rely on unconstrained or softly-penalized optimization, resulting in implausible motions with artifacts such as foot skating and penetration. More importantly, they do not explicitly consider interaction preservation—i.e., maintaining spatial and contact relationship—in the retargeting formulation, relying instead on simple keypoint matching. Consequently, the resulting references are of lower quality, which in turn complicates the downstream RL policy training [12], [8], [13].

In this work, we introduce OMNIRETARGET, an open-source data generation engine that transforms human demonstrations into diverse, high-quality kinematic references for humanoid whole-body control. By modeling spatial and contact relationships between robots, objects, and terrains via an interaction mesh [14], OMNIRETARGET preserves essential interactions and generates kinematically feasible variations. While existing methods require separate demonstrations for each variation—making data collection costly and limiting coverage—OMNIRETARGET addresses this bottleneck directly. Inspired by data augmentation frameworks for contact-rich manipulation [15], our framework automatically augments a single demonstration into a large number of training examples across object configurations, shapes, robot embodiments, and environments.

Our pipeline employs constrained optimization to enforce physical feasibility, including collision avoidance, joint limits, and foot contact stability, while minimizing interaction mesh deformation. The resulting motions are interaction-preserving and exhibit only minimal kinematic artifacts, providing dense learning signals that accelerate RL with minimal reward engineering. On a diverse suite of whole-body interaction tasks such as box lifting, platform climbing, and slope crawling, policies trained on OMNIRETARGET datasets outperform those from prior retargeting methods in both motion quality and robustness, with successful zero-shot sim-to-real transfer onto a physical humanoid robot.

Our contributions are fourfold:

1) The first interaction-preserving humanoid retargeting framework that handles rich robot-object-terrain interactions while enforcing hard physical constraints.
2) A systematic data augmentation pipeline that transforms a single human demonstration into a diverse, large-scale set of high-quality kinematic trajectories on various robot embodiments.
3) A large-scale, open-source dataset of retargeted, kinematically-feasible loco-manipulation trajectories.
4) Successful zero-shot sim-to-real transfer of proprioceptive RL policies on a physical humanoid, demonstrating a diverse set of scene-interaction tasks, including a long, agile sequence of object carrying, platform climbing, jumping, rolling and wall-flipping.

II. RELATED WORKS

A. Motion Retargeting

In computer graphics, transferring motions across characters has been extensively explored. Researchers have employed optimization-based methods to retarget human motions to avatars by preserving distances and orientations between keypoints [16], minimizing deformation energy [14], [17], or scaling the motions to satisfy hard constraints [18]. Others leverage data-driven methods that map diverse skeletons to a canonical representation [19], solve inverse kinematics with neural networks [20], or use reinforcement learning to preserve an interaction graph [21].

Retargeting motions to humanoid robots introduces challenges beyond character animation, particularly the need to enforce physical constraints. For example, PHC [10], a graphics method adopted in robotics [13], [8], uses keypoint matching with unconstrained optimization, often leading to penetration, foot skating, and lack of object or scene awareness. Similarly, GMR [9] extends keypoint matching to orientations but suffer the same issues. VideoMimic [11] improves realism with soft contact and collision penalties but offers no guarantees and requires careful tuning.

The closest method to ours is Interaction Mesh based Motion Adaptation (IMMA) [22], which also leverages an interaction mesh [14] to preserve the spatial relationship between body parts. However, it is not open-sourced and ignores kinematic limits and interactions with the environment or manipulated objects. In contrast, OMNIRETARGET unifies all hard constraints, including foot sticking, non-penetration, and joint and velocity limits, while explicitly preserving environment and object interactions.

B. Learning-Based Humanoid Whole-Body Control

Recent learning-based whole-body control has enabled humanoids to traverse dynamic scenes and manipulate objects [23], [24], [25], [26], [27], [28], [29], [30], [31]. These methods typically train with hand-crafted rewards or task interfaces (e.g., velocity tracking, contact schedules, end-effector targets) but depend on extensive reward engineering and mostly fail to yield natural, human-level motions.

Motion imitation offers a promising alternative. In graphics, DeepMimic [4] shows that using human references yields natural, human-like behaviors with agile, dynamic motions. However, applying this approach to humanoid robots remains difficult due to the lack of reliable open-source kinematic retargeting pipelines. With suboptimal reference motions, practitioners are forced to either manually clean the data [12] or re-introduce extensive reward engineering, such as ad-hoc regularizers for contact, slipping, and air time, to compensate for artifacts [9], [13], [32]. In contrast, trackers with minimal

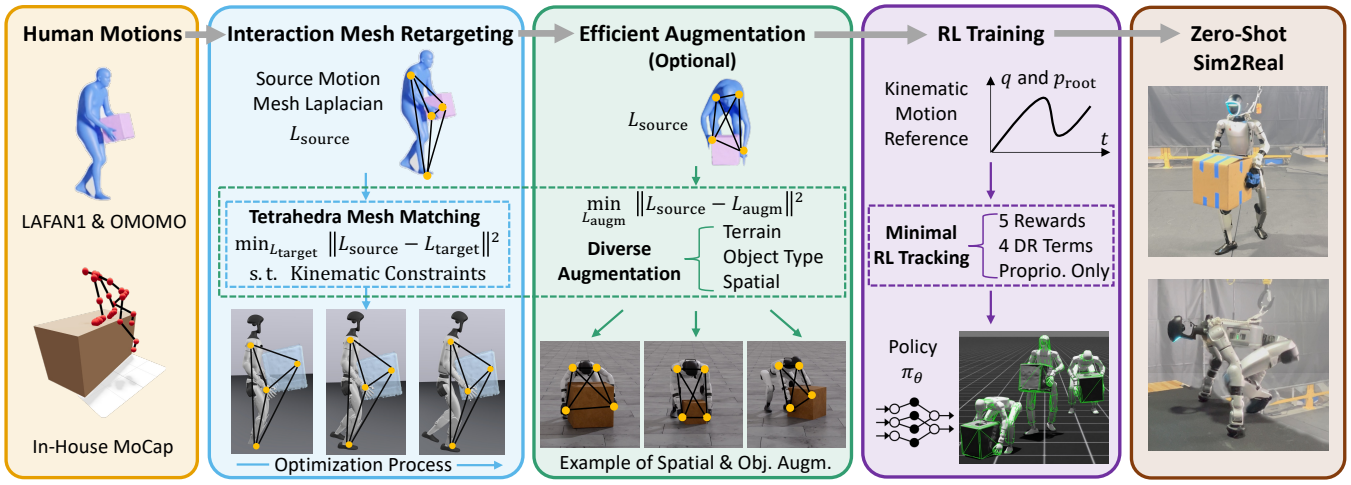

Fig. 2: **OMNIRETARGET overview.** Human demonstrations are retargeted to the robot via interaction-mesh–based constrained optimization. Each spatial and shape augmentation is solved as a new optimization, producing diverse trajectories that serve as references for RL training with minimal reward design and domain randomization, enabling zero-shot transfer to real-world humanoids.

reward formulation like BeyondMimic [33] achieve state-of-the-art results on hardware with high-fidelity references [34], but those are scarce and robot-only, without interactions.

Beyond single-character motion, human–scene interaction data has proven effective for terrain traversal and loco-manipulation in character animation [5], [6], but translating this to robotics remains challenging. VideoMimic [11] applies this idea to human–terrain traversal by reconstructing motions and terrains from video, but suffers from artifacts and is limited to static–scene interactions. To bridge this gap, OMNIRETARGET enables natural, agile robot-object-scene interactions with high-quality reference from retargeting without manual post-processing or reward engineering.

C. Data Generation for Humanoid Loco-Manipulation

The demand for whole-body interaction data has motivated many prior works on data generation. One approach is direct human teleoperation [35], [7], [8], [9], [36]. While it provides online feedback, teleoperation is difficult to scale: it's labor-intensive, prone to operator fatigue, and limited by the embodiment gap between human and robot kinematics. The lack of rich haptic feedback and difficulty stabilizing extreme motions (e.g., deep squats) further constrain its applicability. To address these scaling challenges, automated data augmentation has been explored, particularly for robotic manipulation. Many works leverage state-of-the-art generative models for visual [37], [38], [39] and semantic [40], [41], [42] augmentations, while others rely on simple open-loop kinematic replay of base trajectories [43], [44], [45] or trajectory optimization [15] in simulation. Despite the advances in manipulation, data augmentation for whole-body loco-manipulation remains largely unexplored. The closest prior work [46] interpolates keypoints to augment objects of different shape, but it cannot deal with varied object poses either. OMNIRETARGET directly addresses this gap.

III. INTERACTION-PRESERVING MOTION RETARGETING

A. Interaction Mesh with Hard Constraints

We leverage the interaction mesh [14] to preserve spatial relationships between body parts, objects, and the environment. The interaction mesh is defined as a volumetric structure whose vertices consist of key robot or human joints together with points sampled from objects and the environment. By shrinking or stretching this mesh, we can warp human motion onto the robot while preserving relative spatial configurations and contact relationships.

Interaction Mesh Construction. We construct the interaction mesh by applying Delaunay tetrahedralization [48] to user-defined key joint positions and randomly sampled object and environment points. To more accurately maintain contact relationships, we sample the object and environment surfaces more densely than the body joints.

Optimization Objectives and Constraints. To preserve the spatial relationships between the body parts, objects and terrains, our primary objective is to minimize the Laplacian deformation energy of the interaction meshes [49], [50] constructed from two corresponding sets of keypoints. The source set at frame t, $\mathscr{P}_t^{\text{source}}$, is composed of user-defined anatomical points on the human, and points randomly sampled on the manipulated object and the environment. The target set for the retargeted motion, $\mathscr{P}_t^{\text{target}}$, consists of corresponding anatomical points on the robot, the same manipulated object and environment points. Our method is relatively robust to the precise placement of these keypoints, requiring only a semantically consistent correspondence between the human and robot (e.g., hand to hand).

The Laplacian coordinate of the i-th keypoint $p_{t,i} \in \mathscr{P}_t$ is defined as the difference between the point and the weighted average of its neighbors $j \in \mathscr{N}(i)$:

$$L(p_{t,i}) = p_{t,i} - \sum_{j \in \mathscr{N}(i)} w_{ij} \cdot p_{t,j}, \tag{1}$$

Methods	Hard Kinematic Constraints	Interaction w/ Object	Interaction w/ Terrain	Data Augmentation	Optimization Method
IMMA [22]	✓	✗	✗	✗	QP
PHC [10]	✗	✗	✗	✗	Gradient Descent
GMR [9]	✗	✗	✗	✗	Mink [47]
VideoMimic [11]	Soft Penalty	✗	✓	✗	JAX L-M
OMNIRETARGET (Ours)	✓	✓	✓	✓	Sequential SOCP

TABLE I: Comparison of prior retargeting methods across different aspects.

where w_{ij} is the normalized weight and we omit L's dependence on $\{p_{t,j}\}_{j\neq i}$ in the function definition for conciseness. For all our experiments, we use uniform weights, setting $w_{ij} = 1/|\mathcal{N}(i)|$. The deformation energy measures the change in these Laplacian coordinates between the source demonstration mesh $\mathscr{P}_t^{\text{source}}$ and the retargeted mesh $\mathscr{P}_t^{\text{target}}$:

$$E_L = \sum_{p_{t,i}^{\text{source}} \in \mathscr{P}_t^{\text{source}}, p_{t,i}^{\text{target}} \in \mathscr{P}_t^{\text{target}}} \|L(p_{t,i}^{\text{source}}) - L(p_{t,i}^{\text{target}})\|^2. \quad (2)$$

We seek the robot configuration q_t, consisting of the floating base pose (quaternion and translation) and all joint angles, that minimizes this deformation energy while satisfying a set of hard kinematic constraints. The robot's keypoints are determined by its configuration q_t via forward kinematics f_i as $p_{t,i}^{\text{robot}}(q_t) = f_i(q_t) \in \mathscr{P}_t^{\text{target}}$. At each time step, we solve the following constrained, nonconvex program:

$$q_t^\star = \arg\min_{q_t} \sum_i \|L(p_{t,i}^{\text{source}}) - L(p_{t,i}^{\text{target}}(q_t))\|^2 + \|q_t - q_{t-1}\|_Q^2 \quad (3a)$$

$$\text{s.t.} \quad \phi_j(q_t) \geq 0, \forall j \quad (3b)$$

$$q_{\min} \leq q_t \leq q_{\max} \quad (3c)$$

$$v_{\min} \cdot dt \leq q_t - q_{t-1} \leq v_{\max} \cdot dt \quad (3d)$$

$$p_t^F = p_{t-1}^F, \forall \text{stance foot}, \quad (3e)$$

where Q is a cost matrix that encourages temporal smoothness, ϕ_j denotes the signed distance function for the j-th collision pair, $q_{\min}/q_{\max}$ and $v_{\min}/v_{\max}$ are the configuration and velocity bounds, p_t^F denotes the foot position. A foot is considered to be in the stance phase if its horizontal velocity in the source motion (in the xy-plane) falls below a threshold of 1 cm/s. This optimization program solves for a temporally consistent robot trajectory that minimizes interaction mesh deformation, subject to hard constraints for collision avoidance (3b), joint and velocity limits (3c)–(3d), and preventing foot skating (3e).

We solve (3) sequentially for each timestep using a customized Sequential Quadratic Programming (SQP)-style solver. Within each iteration, the objective (3a) is quadratically approximated and the hard constraints (3b)–(3e) are linearized around the solution from the previous iteration. To ensure temporal consistency and accelerate convergence, the optimization at frame t is warm-started with the optimal solution from the previous frame $q_{t-1}^\star$. A key challenge in this formulation is computing derivatives involving the quaternion-based floating base orientation; our implementation leverages the automatic differentiation framework in Drake [51], which correctly handles the differential geometry of rotations on the $\mathbb{S}^3$ manifold [52].

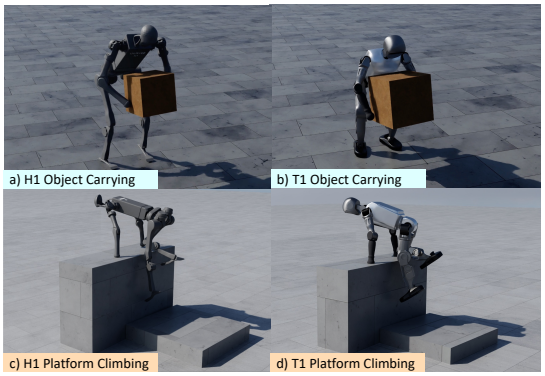

Fig. 3: Cross-embodiment robot-object-terrain interaction.

Our interaction-mesh-based kinematic pipeline is highly general. It adapts to different robot embodiments, including the Unitree G1, H1, and Booster T1 (Fig. 3), by modifying only the keypoint correspondences in the interaction mesh and the robot's collision model. It also supports diverse interaction types: **robot-object** interactions from the OMOMO [1], **robot-terrain** interactions from in-house MoCap data, and **robot-only** motions on flat terrain from LAFAN1 [2].

B. Terrain, Object Shape and Spatial Augmentation

A key advantage of our framework is its capability for systematic data augmentation, which eliminates the need for collecting numerous, repetitive demonstrations with minor spatial variations. Our method can transform a single human demonstration into a rich and diverse dataset by parametrically altering object configurations, shapes, or terrain features. For each new scenario, we re-solve the optimization problem with fixed $\mathscr{P}_t^{\text{source}}$ and augmented $\mathscr{P}_t$: minimizing the interaction mesh deformation finds a new, kinematically valid robot motion $\{q_t\}$ that preserves the essential spatial and contact relationships of the original interaction.

Robot-Object. We generate diverse interactions by augmenting both the object's spatial configuration and its shape. We apply translations and rotations to modify the object's initial pose (Fig. 4b) and blend the new initial pose with the original object motion via interpolation with an exponential schedule detailed in (14). In addition, we scale the three dimensions of the object (Fig. 4c). A critical component of this process is constructing the interaction mesh in the object's local frame, which ensures that the robot's interacting body

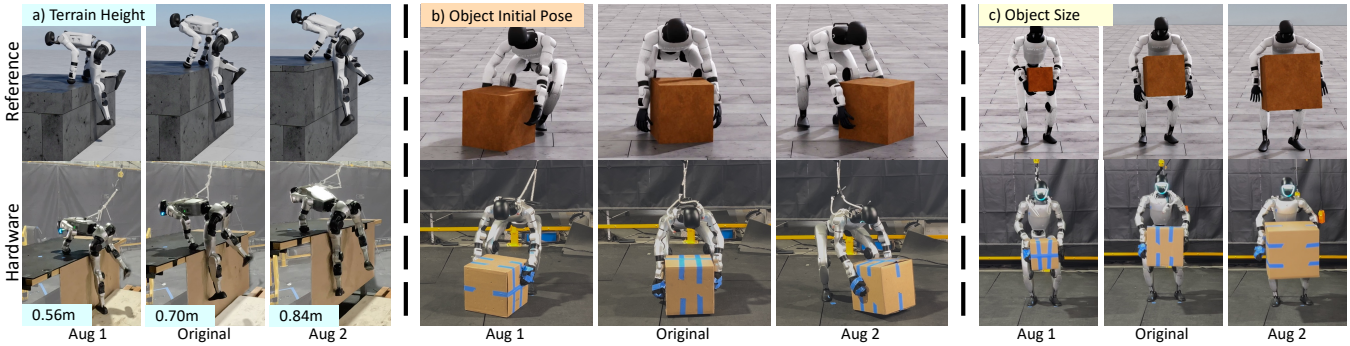

Fig. 4: OMNIRETARGET generates systematic variations of (a) terrain height, (b) object initial pose, and (c) object shape from a single human demonstration, with optimized motions in simulation (top) transferring consistently to hardware (bottom).

parts naturally follow the object's transformation (Sec. VI-C.2).

However, this alone can lead to trivial augmentations where the entire robot undergoes a rigid transformation along with the object. To generate more meaningful data diversity, we introduce cost terms and constraints that anchor parts of the robot's body to the nominal trajectory $\{\bar{q}_t^\star\}$. For instance, in a pick-up task, we encourage the robot to discover new upper-body coordination by penalizing lower-body deviations from the original motion:

$$\|q_t - \bar{q}_t^\star\|_W, \tag{4}$$

where W heavily penalizes the lower-body entries, constraining the initial foot poses to match the nominal trajectory

$$p_0^F = \bar{p}_0^{F\star} \quad \text{for left and right feet.} \tag{5}$$

These added objectives prevent the optimization from collapsing to a simple rigid transform of the nominal trajectory and instead produce genuinely new and diverse interactions.

Robot-Terrain. We generate diverse terrain scenarios by scaling environmental features, such as varying the platform height and depth (Fig. 4a), and introducing additional constraints. For instance, to encourage stable ground contact when the terrain is elevated, we uniformly sample a grid of points on the ground surface into the interaction mesh.

IV. RL TRAINING WITH MINIMAL FORMULATION

Having established our method for generating high-quality kinematic references, we use RL to bridge the gap to dynamics by training a low-level policy that converts these trajectories into physically realizable actions, enabling zero-shot transfer from simulation to hardware.

Reward engineering is often the main difficulty in humanoid RL: prior works [9], [13], [32] rely on many ad-hoc regularizers (e.g., foot flight and contact time) to compensate for artifacts in noisy references, but tuning these terms is tedious and fragile. In contrast, BeyondMimic [33] shows that when references are clean [34], a minimal reward is already sufficient for high-quality tracking. Since OmniRetarget produces such artifact-free, interaction-preserving references, we can follow this minimal formulation directly, achieving faithful tracking of dynamic interactions and zero-shot sim-to-real transfer *without any hyperparameter tuning*.

Observations. To show that high-quality reference motions provide a sufficient prior for complex tasks, we design a *minimal proprioceptive* observation space, as listed below, where the agent is blind to explicit scene and object information and must follow the reference trajectory precisely.

- *Reference Motion:* Reference Joint Position/Velocity, Reference Pelvis Position/Orientation Error;
- *Proprioception:* Pelvis Linear/Angular Velocity, Joint Position/Velocity;
- *Previous Action:* Policy action from last timestep.

For agile motions where state estimation is unreliable, we mask out the pelvis linear position error and velocity.

Rewards. To show the benefits of high-quality reference and avoid reward tuning, we use only five reward terms:

- *Body Tracking:* DeepMimic-style tracking term for body position, orientation, linear and angular velocity;
- *Object Tracking (where applicable):* DeepMimic-style tracking term for object position and orientation;
- *Action Rate:* Penalize rapid changes in action;
- *Soft Joint Limit:* Penalize robot joint limit violation;
- *Self-Collision:* Binary penalty on each body if its self-collision force exceeds 1 N.

We use the same weights and hyperparameters from [33] out of the box without tuning. For object tracking, we use the same hyperparameters as body tracking terms.

Termination. We terminate training episodes with large body tracking deviations [33]. For object loco-manipulation, episodes terminate when the object deviates more than 1.0m and 45° from the reference trajectory. We only apply this criterion after the policy achieves reasonable body tracking.

Domain Randomization. To improve generalization across object properties for a single reference motion, we randomize the object's physical parameters: mass (0.1–2 kg), center of mass (±0.08 m), inertia (50–150%), and shape (±10%). For the robot specifically, in contrast to the many terms in prior works (e.g., random force injection (RFI), motor PD, action delay), we only apply four terms:

- *Torso COM Position*: ±0.025 m in x, ±0.05 m in y, ±0.075 m in z;
- *Joint default position*: ±0.01 rad;
- *Random push*: 0.3 m/s, 0.78 rad/s for (1–3) s;

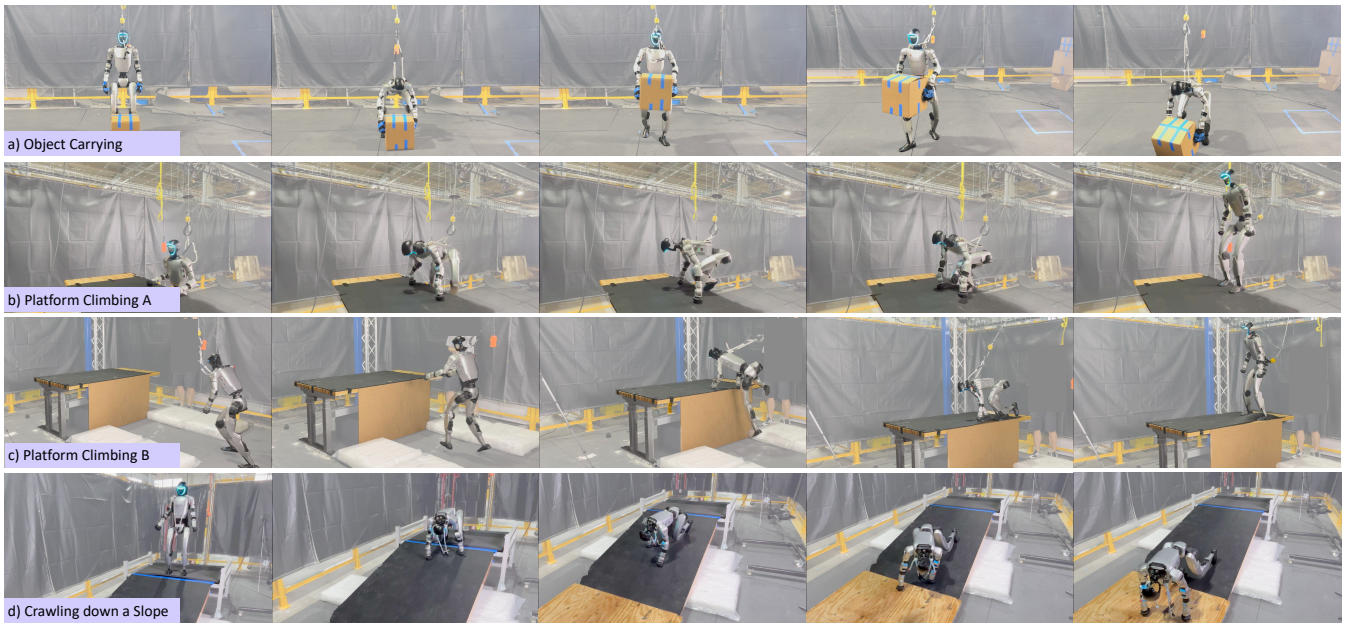

Fig. 5: Additional hardware results showing diverse, agile and human-like behaviors.

- *Observation noise*: ± 0.05 for orientation in Rot6D, ± 0.5 m/s and ± 0.2 rad/s for linear and angular velocity, ± 0.01 rad and ± 0.5 rad/s for joint position and velocity.

Policy Training. We group similar motions for faster training. All box-moving motions share a single multi-task policy, while platform climbing uses one policy per reference.

V. Experimental Results

In this section, we present a comprehensive experimental validation of OmniRetarget. We first demonstrate the breadth of complex behaviors enabled by our approach, including natural object manipulation and terrain interaction. We then provide a quantitative benchmark against state-of-the-art baselines, evaluating performance across kinematic quality metrics and downstream policy performance.

A. Whole-Body Scene-Object-Interaction

a) Agile Loco-Manipulation: OmniRetarget enables RL policies to learn agile, whole-body motions for complex scene interactions and loco-manipulation in simulation, culminating in successful zero-shot sim-to-real transfer to hardware. Shown in Fig. 5, policies trained on our data reproduce a diverse range of expressive behaviors on a Unitree G1 humanoid, including natural box-carrying motions retargeted from the OMOMO dataset, dynamically climbing a 0.9m-high platform (70% of the robot's height), and crawling over slopes, showing clean and accurate contact sequences.

To showcase the full capabilities of our framework, we present a long-horizon, dynamic sequence inspired by the Boston Dynamics Atlas tool-use demo [53]. Visualized in Fig. 1, our retargeted data enables the robot to carry a 4.6 kg chair to a platform, use it as a stepstone to climb up, and then leap off, performing a parkour-style roll to absorb the landing impact. This 30-second, complex, multi-stage task highlights OmniRetarget's ability to produce precise and versatile reference motions, pushing the boundaries of what is possible for humanoids learning agile, human-like behaviors.

We additionally showcase a high-dynamic wall-flip motion[1] in Fig. 6. The robot completes the full flip in approximately 0.5 second, reaching a peak angular velocity of 15 rad/s. Unlike the human foot, which can flex at the arch to maintain contact and generate friction, the robot foot is rigid. As a result, it must align more closely to the wall to achieve sufficient contact area and friction. To account for this physical difference and give RL more freedom to learn this skill, we relaxed the termination condition during RL training by increasing the end-effector position error threshold to 0.5 meter (compared to 0.25 meter used in other motions) and removed the foot joint orientation tracking term from the reward function. All other components of the tracking objective remain consistent with other motions. The trained policy is robust and achieves a $5/5$ success rate in our real-world experiments.

b) Sim-to-real with Augmented Data: We show that the augmented motions from our pipeline can be used for training and deployment effectively. As shown in Fig. 4, the interaction mesh formulation allows OmniRetarget to generalize a single nominal motion into box-picking across shapes and positions, as well as platform climbing at different heights. Notably, these augmented motions transfer to hardware without reward tuning, effectively expanding the repertoire of scenes and behaviors we can achieve in real.

In comparison, relying solely on domain randomization–which perturbs object shapes and poses only during training–performs poorly under our RL formulation, as the policies struggle to explore far beyond the nominal reference. Policies

[1]The motion is acquired from `https://actorcore.reallusion.com/3d-motion?asset=parkour-tic-tac-backflip` . An IMU capable of measuring angular rates above 15 rad/s is required for this motion.

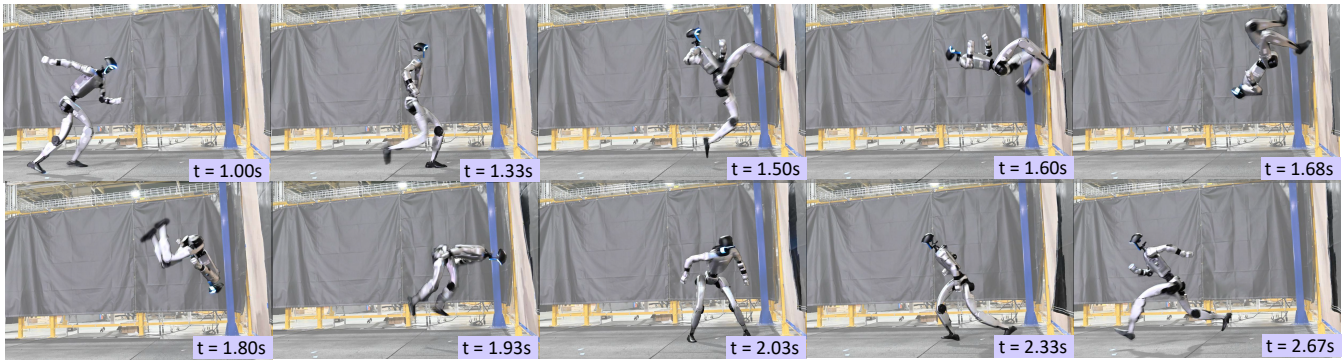

Fig. 6: Hardware results showing a high-dynamic wall-flip motion. The robot reaches a maximum linear velocity of 3.5 m/s and a peak angular velocity of 15 rad/s.

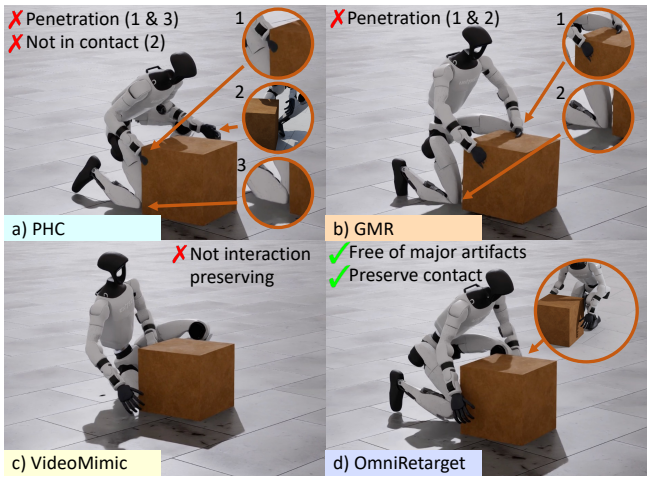

Fig. 7: Artifacts resulting from the retargeting baselines.

trained on our augmented data instead yield reliable success (see video for comparison). Admittedly, additional reward engineering could help, but it contradicts our minimal design goal. Quantitatively, training and evaluating on the full augmented dataset achieves a 79.1% success rate, comparable to 82.2% when evaluating on nominal motions only, showing that kinematics augmentation substantially enlarges coverage without significant performance degradation.

B. Benchmark Against Prior Retargeting Pipelines

We compare OMNIRETARGET against several widely-used open-source retargeting baselines[2]: PHC [10], GMR [9] and VideoMimic [11]. The generated dataset including 2.78 hours of box carrying in OMOMO, 1 hour of in-house MoCap and 4.6 hours of LAFAN1 will be open-sourced.

a) Kinematics Quality: We evaluate the kinematic quality of retargeted motions on a Unitree G1 with three criteria:

1) **Penetration**: Measured by the time duration (normalized by the trajectory length) and maximum depth of intersections between the robot, objects, and terrain.
2) **Foot Skating**: Quantified by the time duration (normalized by the total desired foot sticking length) and maximum skating velocity of a stance foot.
3) **Contact Preservation**: Quantified by the time duration (normalized by the desired contact length). For *robot-object* tasks, we measure hand-object contact. For *robot-terrain* tasks, we measure contact between the robot's hands, toes, and heels with the terrain surface.

As illustrated in Tab. II, OMNIRETARGET significantly outperforms all baselines across most kinematic metrics. While OMNIRETARGET occasionally incurs minor penetration due to the linearization of constraints (3b) in the sequential SOCP solver, the violations are minimal and can be efficiently fixed by RL. GMR achieves the highest contact preservation score for robot-object interaction tasks; however, this outcome largely reflects its keypoint-matching objective. In practice, scaling human hand keypoints to the robot's size often drives the robot's hands inside the object, leading to substantial penetration errors (Fig. 7b). Overall, all baselines exhibit significant penetration and foot skating (Fig. 7), degrading the downstream RL performance, as discussed next.

For a direct comparison on pure locomotion, we retarget motions from the LAFAN1 MoCap dataset [2] and benchmark them against the publicly available Unitree LAFAN1 retargeted dataset [34]. This serves as a strong baseline, as it is widely considered a high-quality data source for RL-based locomotion training [33]. Table II shows that OMNIRETARGET's motions exhibit fewer physical artifacts, achieving better satisfaction of hard constraints.

b) Downstream RL Performance: A central observation from prior works [33], [12] is that the quality of retargeted motions strongly influences the performance of downstream RL. To verify this, we select 39 challenging motions for OMNIRETARGET and baselines, and train RL policies using identical hyperparameters from [33] without manual tuning. We evaluate the policies in simulation, and success is measured by training termination criteria.

Shown in Tab. II, retargeting quality directly impacts RL success rates. OMNIRETARGET consistently achieves the highest performance across tasks, exceeding baselines by over 10% with lower variance, which indicates more stable learning across different motions. PHC performs better than

[2]Baseline performance may depend on their hyperparameters. We initialized from the default settings in their public codes, and further improved to ensure consistent performance across different tasks.

Method	Penetration		Foot Skating		Contact Preservation	Downstream RL Policy
	Duration ↓	Max Depth (cm) ↓	Duration ↓	Max Vel. (cm/s) ↓	Duration ↑	Success Rate ↑
Robot-Object Interaction (Retargeting from the OMOMO Dataset)						
PHC [10]	0.68 ± 0.21	5.11 ± 3.09	0.05 ± 0.05	1.40 ± 0.80	0.96 ± 0.09	71.28% ± 22.55%
GMR [9]	0.83 ± 0.14	8.50 ± 3.94	0.02 ± 0.01	1.46 ± 0.45	**0.99 ± 0.04**	50.83 % ± 23.89%
VideoMimic [11]	0.60 ± 0.27	7.48 ± 4.95	0.12 ± 0.07	1.50 ± 0.70	0.77 ± 0.25	3.85% ± 8.41%
OMNIRETARGET	**0.00 ± 0.01**	**1.34 ± 0.34**	**0**	**0**	0.96 ± 0.09	**82.20% ± 9.74%**
Robot-Terrain Interaction (Retargeting from the In-House MoCap Dataset)						
PHC	0.66 ± 0.36	7.74 ± 4.53	0.15 ± 0.04	2.03 ± 1.83	0.45 ± 0.28	52.63% ± 49.93%
GMR	0.91 ± 0.16	5.72 ± 3.84	0.04 ± 0.05	1.75 ± 3.01	0.67 ± 0.26	78.94% ± 40.77%
VideoMimic	0.83 ± 0.11	5.97 ± 3.58	0.14 ± 0.05	1.85 ± 1.38	0.47 ± 0.25	51.75% ± 49.23%
OMNIRETARGET	**0.01 ± 0.02**	**1.37 ± 0.18**	**0**	**0**	**0.72 ± 0.19**	**94.73% ± 22.33%**
Robot-Only (Retargeting from the LAFAN1 Dataset)						
Unitree [34]	0.09 ± 0.13	3.22 ± 2.64	0.06 ± 0.03	1.46 ± 0.01	N/A	**100%**
OMNIRETARGET	**0.00 ± 0.00**	**1.07 ± 0.00**	**0**	**0**	N/A	**100%**

TABLE II: Quantitative comparison of kinematic retargeting quality and downstream RL performances.

GMR in object manipulation, likely due to lower penetration with sufficient contact preservation, but worse in terrain interaction, where its contact preservation drops by nearly 50%. Specifically for terrain interaction, we see that contact preservation is directly proportional to the success rate. These results suggest that both contact preservation and penetration reduction are critical for generalizing RL policies across diverse tasks, and OMNIRETARGET shows strength in both.

VideoMimic shows the weakest interaction preservation among all baselines (Fig. 7c), likely due to its collision avoidance soft cost conflicting with the keypoint matching cost. This is compounded by its coarse collision model originally designed for heightmaps, which is ill-suited for precise loco-manipulation. Consequently, while its terrain-interaction results are comparable to PHC, its performance on object manipulation is poor. Although this could be partially attributed to the tuning of its soft penalties, OMNIRETARGET demonstrates that a hard-constraint formulation avoids such sensitivities altogether.

VI. CONCLUSION

In this work, we tackled a key data bottleneck caused by a lack of high-quality, interaction-aware retargeting pipeline in humanoid whole-body loco-manipulation. We introduced OMNIRETARGET, a unified, interaction-preserving data generation engine that leverages an interaction mesh within a single constrained optimization. Our experiments showed that OMNIRETARGET significantly outperforms prior methods in kinematic quality, producing a diverse set of artifact-free trajectories from single demonstrations. This high-quality data enabled a proprioceptive RL policy, trained with minimal formulation, to achieve long-horizon dynamic skills on a physical humanoid via zero-shot sim-to-real transfer.

Ultimately, OMNIRETARGET demonstrates a paradigm shift from patching lower-quality reference motions with complex reward engineering to solving the problem at its source with principled data generation. While our current frame-by-frame optimization is mostly effective, future work could explore jointly optimizing the entire trajectory to enhance the framework's robustness to noisier motion sources, such as video data, or learning autonomous visuomotor policies. By open-sourcing our complete framework and the large-scale dataset of retargeted trajectories, we hope to accelerate progress towards more agile, capable, and versatile humanoid robots.

REFERENCES

[1] J. Li, J. Wu, and C. K. Liu, "Object motion guided human motion synthesis," *ACM Transactions on Graphics (TOG)*, 2023.

[2] F. G. Harvey, M. Yurick, D. Nowrouzezahrai, and C. Pal, "Robust motion in-betweening," vol. 39, no. 4, 2020.

[3] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Science robotics*, 2020.

[4] 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 (TOG)*, 2018.

[5] M. Xu, Y. Shi, K. Yin, and X. B. Peng, "Parc: Physics-based augmentation with reinforcement learning for character controllers," in *Proceedings of the SIGGRAPH Conference Papers*, 2025.

[6] Z. Wu, J. Li, P. Xu, and C. K. Liu, "Human-object interaction from human-level instructions," *arXiv preprint arXiv:2406.17840*, 2024.

[7] Z. Fu, Q. Zhao, Q. Wu, G. Wetzstein, and C. Finn, "Humanplus: Humanoid shadowing and imitation from humans," *CoRL*, 2024.

[8] T. He, Z. Luo, X. He, W. Xiao, C. Zhang, W. Zhang, K. M. Kitani, C. Liu, and G. Shi, "Omnih2o: Universal and dexterous human-to-humanoid whole-body teleoperation and learning," in *CoRL*, 2025.

[9] Y. Ze, Z. Chen, J. P. AraÃšjo, Z.-a. Cao, X. B. Peng, J. Wu, and C. K. Liu, "Twist: Teleoperated whole-body imitation system," *CoRL*, 2025.

[10] Z. Luo, J. Cao, A. W. Winkler, K. Kitani, and W. Xu, "Perpetual humanoid control for real-time simulated avatars," in *ICCV*, 2023.

[11] A. Allshire, H. Choi, J. Zhang, D. McAllister, A. Zhang, C. M. Kim, T. Darrell, P. Abbeel, J. Malik, and A. Kanazawa, "Visual imitation enables contextual humanoid control," *CoRL*, 2025.

[12] T. Zhang, B. Zheng, R. Nai, Y. Hu, Y.-J. Wang, G. Chen, F. Lin, J. Li, C. Hong, K. Sreenath *et al.*, "Hub: Learning extreme humanoid balance," *CoRL*, 2025.

[13] T. He, J. Gao, W. Xiao, Y. Zhang, Z. Wang, J. Wang, Z. Luo, G. He, N. Sobanbabu, C. Pan, Z. Yi, G. Qu, K. Kitani, J. Hodgins, L. J. Fan, Y. Zhu, C. Liu, and G. Shi, "Asap: Aligning simulation and real-world physics for learning agile humanoid whole-body skills," *arXiv*, 2025.

[14] E. S. L. Ho, T. Komura, and C.-L. Tai, "Spatial relationship preserving character motion adaptation," *ACM Transactions on Graphics*, 2010.

[15] L. Yang, H. Suh, T. Zhao, B. P. Graesdal, T. Kelestemur, J. Wang, T. Pang, and R. Tedrake, "Physics-driven data generation for contact-rich manipulation via trajectory optimization," *RSS*, 2025.

[16] T. Cheynel, T. Rossi, B. Bellot-Gurlet, D. Rohmer, and M.-P. Cani, "Sparse motion semantics for contact-aware retargeting," in *ACM SIGGRAPH Conference on Motion, Interaction and Games*, 2023.

[17] Y. Kim, H. Park, S. Bang, and S.-H. Lee, "Retargeting human-object interaction to virtual avatars," *IEEE transactions on visualization and computer graphics*, vol. 22, no. 11, pp. 2405–2412, 2016.

[18] M. Gleicher, "Retargetting motion to new characters," in *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, 1998, pp. 33–42.
[19] K. Aberman, P. Li, D. Lischinski, O. Sorkine-Hornung, D. Cohen-Or, and B. Chen, "Skeleton-aware networks for deep motion retargeting," *ACM Transactions on Graphics (TOG)*, vol. 39, no. 4, pp. 62–1, 2020.
[20] R. Villegas, J. Yang, D. Ceylan, and H. Lee, "Neural kinematic networks for unsupervised motion retargetting," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018.
[21] Y. Zhang, D. Gopinath, Y. Ye, J. Hodgins, G. Turk, and J. Won, "Simulation and retargeting of complex multi-character interactions," in *ACM SIGGRAPH 2023 Conference Proceedings*, 2023.
[22] S. Nakaoka and T. Komura, "Interaction mesh based motion adaptation for biped humanoid robots," in *Humanoids*, 2012.
[23] J. Dao, H. Duan, and A. Fern, "Sim-to-real learning for humanoid box loco-manipulation," in *ICRA*. IEEE, 2024.
[24] J. Long, J. Ren, M. Shi, Z. Wang, T. Huang, P. Luo, and J. Pang, "Learning humanoid locomotion with perceptive internal model," *arXiv*, 2024.
[25] J. He, C. Zhang, F. Jenelten, R. Grandia, M. Bächer, and M. Hutter, "Attention-based map encoding for learning generalized legged loco-motion," *Science Robotics*, vol. 10, no. 105, p. eadv3604, 2025.
[26] X. He, R. Dong, Z. Chen, and S. Gupta, "Learning getting-up policies for real-world humanoid robots," *RSS*, 2025.
[27] Y. Kuang, H. Geng, A. Elhafsi, T.-D. Do, P. Abbeel, J. Malik, M. Pavone, and Y. Wang, "Skillblender: Towards versatile humanoid whole-body loco-manipulation via skill blending," *arXiv*, 2025.
[28] Z. Zhang, C. Chen, H. Xue, J. Wang, S. Liang, Y. Liu, Z. Zhang, H. Wang, and L. Yi, "Unleashing humanoid reaching potential via real-world-ready skill space," *arXiv preprint arXiv:2505.10918*, 2025.
[29] Y. Xue, W. Dong, M. Liu, W. Zhang, and J. Pang, "A unified and general humanoid whole-body controller for versatile locomotion," *RSS*, 2025.
[30] C. Zhang, W. Xiao, T. He, and G. Shi, "Wococo: Learning whole-body humanoid control with sequential contacts, 2024," *arXiv*, 2024.
[31] Y. Zhang, Y. Yuan, P. Gurunath, T. He, S. Omidshafiei, A.-a. Agha-mohammadi, M. Vazquez-Chanlatte, L. Pedersen, and G. Shi, "Falcon: Learning force-adaptive humanoid loco-manipulation," *arXiv*, 2025.
[32] Z. Li, X. B. Peng, P. Abbeel, S. Levine, G. Berseth, and K. Sreenath, "Reinforcement learning for versatile, dynamic, and robust bipedal locomotion control," *IJRR*, 2025.
[33] Q. Liao, T. E. Truong, X. Huang, G. Tevet, K. Sreenath, and C. K. Liu, "Beyondmimic: From motion tracking to versatile humanoid control via guided diffusion," *arXiv e-prints*, pp. arXiv–2508, 2025.
[34] U. Robotics and Contributors, "Unitree lafan1 retargeting dataset," https://huggingface.co/datasets/lvhaidong/LAFAN1_Retargeting_Dataset, 2025.
[35] M. Seo, S. Han, K. Sim, S. H. Bang, C. Gonzalez, L. Sentis, and Y. Zhu, "Deep imitation learning for humanoid loco-manipulation through human teleoperation," in *Humanoids*, 2023.
[36] Q. Ben, F. Jia, J. Zeng, J. Dong, D. Lin, and J. Pang, "Homie: Humanoid loco-manipulation with isomorphic exoskeleton cockpit," *RSS*, 2025.
[37] X. Zhang, M. Chang, P. Kumar, and S. Gupta, "Diffusion meets dagger: Supercharging eye-in-hand imitation learning," in *RSS*, 2024.
[38] S. Tian, B. Wulfe, K. Sargent, K. Liu, S. Zakharov, V. C. Guizilini, and J. Wu, "View-invariant policy learning via zero-shot novel view synthesis," in *CoRL*, 2025.
[39] L. Y. Chen, C. Xu, K. Dharmarajan, Z. Irshad, R. Cheng, K. Keutzer, M. Tomizuka, Q. Vuong, and K. Goldberg, "Rovi-aug: Robot and viewpoint augmentation for cross-embodiment robot learning," in *Conference on Robot Learning (CoRL)*, 2024.
[40] Z. Mandi, H. Bharadhwaj, V. Moens, S. Song, A. Rajeswaran, and V. Kumar, "Cacti: A framework for scalable multi-task multi-scene visual imitation learning," *arXiv preprint arXiv:2212.05711*, 2022.
[41] Z. Chen, S. Kiami, A. Gupta, and V. Kumar, "Genaug: Retargeting behaviors to unseen situations via generative augmentation," *RSS*, 2023.
[42] T. Yu, T. Xiao, A. Stone, J. Tompson, A. Brohan, S. Wang, J. Singh, C. Tan, J. Peralta, B. Ichter *et al.*, "Scaling robot learning with semantically imagined experience," *RSS*, 2023.
[43] A. Mandlekar, S. Nasiriany, B. Wen, I. Akinola, Y. Narang, L. Fan, Y. Zhu, and D. Fox, "Mimicgen: A data generation system for scalable robot learning using human demonstrations," in *CoRL*, 2023.
[44] Z. Jiang, Y. Xie, K. Lin, Z. Xu, W. Wan, A. Mandlekar, L. Fan, and Y. Zhu, "Dexmimicgen: Automated data generation for bimanual dexterous manipulation via imitation learning," *ICRA*, 2025.
[45] C. Garrett, A. Mandlekar, B. Wen, and D. Fox, "Skillmimicgen: Automated demonstration generation for efficient skill learning and deployment," in *Conference on Robot Learning (CoRL)*, 2024.
[46] S. Starke, H. Zhang, T. Komura, and J. Saito, "Neural state machine for character-scene interactions," *ACM Transactions on Graphics*, vol. 38, no. 6, p. 178, 2019.
[47] K. Zakka, "Mink: Python inverse kinematics based on MuJoCo," May 2025. [Online]. Available: https://github.com/kevinzakka/mink
[48] H. Si and K. Gärtner, "Meshing piecewise linear complexes by constrained delaunay tetrahedralizations," in *Proceedings of the 14th international meshing roundtable*. Springer, 2005, pp. 147–163.
[49] M. Alexa, "Differential coordinates for local mesh morphing and deformation," *The Visual Computer*, vol. 19, no. 2, pp. 105–114, 2003.
[50] K. Zhou, J. Huang, J. Snyder, X. Liu, H. Bao, B. Guo, and H.-Y. Shum, "Large mesh deformation using the volumetric graph laplacian," in *ACM SIGGRAPH 2005 Papers*. ACM, 2005, pp. 496–503.
[51] R. Tedrake and the Drake Development Team, "Drake: Model-based design and verification for robotics," 2019.
[52] B. E. Jackson, K. Tracy, and Z. Manchester, "Planning with attitude," *IEEE Robotics and Automation Letters*, 2021.
[53] Boston Dynamics, "Atlas Gets a Grip," YouTube, available: https://www.youtube.com/watch?v=-e1_QhJ1EhQ.
[54] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "SMPL: A skinned multi-person linear model," *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, vol. 34, no. 6, pp. 248:1–248:16, Oct. 2015.

APPENDIX

A. Different Sources of Human Motion Data

Human motion datasets contain rich pose and shape information, but they differ both in data format and in the physical attributes (e.g., height, body proportions) of the demonstrators. To make them compatible across different sources and suitable for retargeting, we need to convert these inputs into a consistent representation, typically a time series of global 3D keypoint positions $\{p^{\text{source}}_{0:T,i}\}$. This process must account for differences between human demonstrators and the target robot.

The datasets used in this work represent two common formats:

- **Parametric Human Models**: The OMOMO dataset uses the SMPL format [54], a parametric model representing human body shape and pose-dependent variations using shape (β) and pose (q) parameters.
- **Skeleton Hierarchy**: Both our in-house MoCap data and the LAFAN1 dataset utilize the skeleton hierarchy defined in the BVH format.

Different retargeting pipelines use different strategies to handle these formats. We detail these preprocessing steps below, denoting the human demonstrator's pose as q^{demo}_t, the SMPL forward model for the i-th keypoint as M_i, the original demonstrator shape as β^{demo}, and the demonstrator's i-th keypoint position as $p^{\text{demo}}_{t,i}$.

1) SMPL Data: To handle data from parametric models like SMPL, methods typically follow one of two strategies: fitting the model to the robot's morphology or directly scaling the human's keypoints.

a) Model Fitting (PHC, VideoMimic): This strategy fits a scaled SMPL model to the robot's morphology. PHC first optimizes for an overall scaling factor α, and a set of SMPL shape parameters β that best match the robot's link length in a canonical T-pose, as detailed in Alg. 1. The final source keypoint positions are then generated from this fitted model:

$$p^{\text{source}}_{t,i} = \alpha \cdot M_i(q^{\text{demo}}_t; \beta). \tag{6}$$

Algorithm 1 Fit SMPL Shape (PHC)

Require: SMPL model M, robot urdf with forward kinematics f, $\bar{q}^{\text{smpl}} = 0_{n_s}, \bar{q}^{\text{robot}} = 0_{n_x}$
Ensure: scaling factors α, β

$\alpha, \beta \leftarrow 1, 0_{10}$
for iter $= 1, \ldots,$ max_iter **do**
  $L(\alpha, \beta) = \sum_i \left\| (f_i(\bar{q}^{\text{robot}}) - \alpha \cdot M_i(\bar{q}^{\text{smpl}}; \beta) \right\|^2$
  $\alpha \leftarrow \alpha - \eta_\alpha \cdot \nabla_\alpha L$
  $\beta \leftarrow \beta - \eta_\beta \cdot \nabla_\beta L$
end for

VideoMimic adopts a similar philosophy but integrates the scaling directly into its main retargeting optimization, solving for per-link scale factors jointly with the robot's motion.

b) Direct Scaling (GMR & OMNIRETARGET*):* In contrast, GMR and OMNIRETARGET use a more direct approach. They generate keypoints from the human's *original* SMPL parameters β^{demo} and then scale them to the robot's proportions. Both methods support detailed morphological adaptation via per-bone scaling factors based on corresponding human-robot link lengths. For simplicity in this work, however, we adopt a single global scaling factor α, set to the robot-to-human height ratio:

$$p_{t,i}^{\text{source}} = \alpha \cdot M_i(q_t^{\text{demo}}; \beta^{\text{demo}}), \alpha = \frac{h_{\text{robot}}}{h_{\text{demo}}}. \tag{7}$$

2) Skeleton Hierarchy Data: For formats like BVH, keypoint positions are derived from the skeleton's forward kinematics f^{skeleton}. This data is then typically scaled to the robot's size using the height ratio:

$$p_{t,i}^{\text{source}} = \frac{h_{\text{robot}}}{h_{\text{demo}}} \cdot f_i^{\text{skeleton}}(q_t^{\text{demo}}). \tag{8}$$

A key distinction among methods is their data compatibility. While GMR and OMNIRETARGET are designed to process both parametric model data and raw skeleton hierarchies, frameworks like PHC and VideoMimic are primarily designed for SMPL data. Fitting other data formats to the SMPL format is yet another tedious process.

B. Different Kinematic Retargeting Formulations

Once human motion is preprocessed into a series of source keypoint positions $\{p_{0:T,i}^{\text{source}}\}$, different methods formulate the retargeting problem in distinct ways. As summarized in Tab. III, these approaches vary in their optimization strategy and objectives. The following sections detail the mathematical formulation of each baseline method and our proposed approach, OMNIRETARGET.

1) PHC: PHC formulates retargeting as a large-scale trajectory-wise optimization problem. It applies gradient descent to minimize the error between the source keypoint positions and the robot's keypoint positions over the entire trajectory, as shown in Alg. 2.

2) GMR: GMR performs retargeting by solving an inverse kinematics (IK) problem at each frame (3). At each timestep, GMR finds the robot configuration q_t that matches the source keypoint positions and orientations via the following optimization program:

$$\begin{aligned} q_t^\star = \arg\min_{q_t} & \sum_i \left\| f_i^p(q_t) - p_{t,i}^{\text{source}} \right\|^2 + \left\| f_i^\theta(q_t) - \theta_{t,i}^{\text{source}} \right\|^2 \\ & \text{s.t. } q_{\min} \le q_t \le q_{\max}, \end{aligned} \tag{9}$$

where f_i^p and f_i^θ are the robot forward kinematics for the i-th keypoint's position and orientation, respectively. Leveraging the mink [47] library, GMR solves this program in a Sequential Quadratic Programming fashion.

Algorithm 2 Retarget Robot Motion (PHC)

Require: Robot urdf, source keypoint positions $\{p_{0:T,i}^{\text{source}}\}$
Ensure: $q_{0:T}$

$q_{0:T} \leftarrow [0_{n_x}]_T$
for iter $= 1, \ldots,$ max_iter **do**
  $\mathscr{L}(q_{0:T}) = \sum_{t=0}^{T} \sum_i \left\| f_i(q_t) - p_{t,i}^{\text{source}} \right\|^2$
  $q_{0:T} \leftarrow \text{clamp}(q_{0:T} - \nabla_{q_{0:T}} \mathscr{L}(q_{0:T}), q_{\min}, q_{\max})$
end for

Algorithm 3 Retarget Robot Motion (GMR)

Require: Robot urdf, source keypoint positions $\{p_{0:T,i}^{\text{source}}\}$ and orientations $\{\theta_{0:T,i}^{\text{source}}\}$
Ensure: $q_{0:T}$

for $t = 0, \ldots, T$ **do**
  $q_t \leftarrow$ Solve IK (9)
end for

3) VideoMimic: VideoMimic jointly optimizes for the robot motion $q_{0:T}$ and SMPL per-link scaling factor β over the entire trajectory. The primary objective is to preserve the scaled pairwise distance and orientation between each keypoint pair (i, j):

$$\mathscr{L}_{\text{pairwise}} = \sum_{t, i \in \mathscr{N}(j)} \| \beta_{ij} \cdot (p_{t,i}^{\text{demo}} - p_{t,j}^{\text{demo}}) - (f_i(q_t) - f_j(q_t)) \|_2^2, \tag{10}$$

with soft penalties on foot contact matching $\mathscr{L}_{\text{contact}}$, foot skating $\mathscr{L}_{\text{skating}}$, collision $\mathscr{L}_{\text{collision}}$, joint limits $\mathscr{L}_{\text{joint}}$ and temporal smoothness $\mathscr{L}_{\text{smooth}}$:

$$\begin{aligned} q_{0:T}^\star, \beta^\star = \arg\min_{q_{0:T}, \beta} \quad & \mathscr{L}_{\text{pairwise}} + \lambda_c \cdot \mathscr{L}_{\text{contact}} + \lambda_s \cdot \mathscr{L}_{\text{skate}} + \\ & \lambda_{cl} \cdot \mathscr{L}_{\text{collision}} + \lambda_j \cdot \mathscr{L}_{\text{joint}} + \lambda_{sm} \cdot \mathscr{L}_{\text{smooth}} + \ldots \end{aligned} \tag{11}$$

Algorithm 4 Retarget Robot Motion (VideoMimic)

Require: Robot urdf, demonstrator's original keypoint positions $\{p_{0:T,i}^{\text{demo}}\}$
Ensure: $q_{0:T}, \beta \leftarrow$ Solve (11) for the entire trajectory

4) IMMA Multi-stage Optimization: IMMA relies on a complex, multi-stage pipeline: first, it optimizes the intermediate robot keypoint positions to warp the interaction mesh

Method	Optimization Type	Primary Objective	Preprocessing	Data Formats
PHC	Trajectory-wise Optimization	Keypoint Position Matching	Model Fitting	SMPL
GMR	Per-Frame Optimization	Keypoint Position & Orientation Matching	Direct Scaling	SMPL, BVH
VideoMimic	Trajectory-wise Optimization	Pairwise Distance & Orientation Preservation	Model Fitting	SMPL
IMMA	Multi-Stage Trajectory-wise Optimization	Interaction Mesh Deformation + IK	Unknown	Unknown
OMNIRETARGET	Per-Frame Optimization	Interaction Mesh Deformation	Direct Scaling	SMPL, BVH

TABLE III: Comparison of different retargeting methodologies

from the human to the robot with minimal deformation by solving the following program

$$\begin{aligned} p^{\star}_{t,i} = \arg\min_{p_{t_i}} \quad & \sum_i \|L(p^{\text{source}}_{t,i}) - L(p_{t,i})\|^2 \\ \text{s.t.} \quad & \phi_j(q_t) \geq 0, \forall j \\ & \|p_{t,i} - p_{t,j}\|_2 = l_{ij}, \forall \text{bone} \\ & p^F_t = p^F_{t-1}, \forall \text{stance foot}, \end{aligned} \tag{12}$$

where l_{ij} is the bone length between the i-th and j-th joints. Then, it solves a separate IK problem to recover joint angles that best match the intermediate keypoints:

$$q^{\star}_t = \arg\min_{q_t} \sum_i \|f_i(q_t) - p^{\star}_{t,i}\|^2_2. \tag{13}$$

In later stages, additional hard constraints on the feet and waist are imposed to prevent foot slipping and ensure dynamic balancing. This sequential and fragmented approach produces dynamically consistent motions but fails to consider crucial kinematic constraints like joint and velocity limits.

Algorithm 5 Retarget Robot Motion (OMNIRETARGET)

Require: Robot urdf, source keypoint positions $\{p^{\text{source}}_{0:T,i}\}$
Ensure: $q_{0:T}$
1: **for** $t = 0, \ldots, T$ **do**
2: $q_t \leftarrow$ Solve interaction mesh optimization (3)
3: **end for**

5) OMNIRETARGET*:* OMNIRETARGET, as outlined in Alg. 5, operates frame-by-frame by minimizing the Laplacian deformation of the interaction meshes. The core objective (3a) is flexible and can be augmented with task-specific costs, such as the orientation matching term from GMR, providing a unified and extensible framework for motion retargeting.

C. Data Augmentation Details

1) Augmented Object Trajectory: To generate a perturbed object trajectory, we introduce a transient offset that decays exponentially over time. Let the original trajectory be denoted by $(p_{obj}(t), \theta_{obj}(t))$. We define an initial positional offset Δp_{obj} and rotational offset $\Delta\theta_{obj}$ that are applied at the onset of object motion, t_m. The augmented trajectory, $(\tilde{p}_{obj}(t), \tilde{\theta}_{obj}(t))$, is then formulated as:

$$\tilde{p}_{obj}(t) = \begin{cases} \Delta p_{obj} + p_{obj}(0) & \text{if } t < t_m \\ \Delta p_{obj}\, e^{-(t-t_m)/\tau_p} + p_{obj}(t) & \text{if } t \geq t_m \end{cases} \tag{14a}$$

$$\tilde{\theta}_{obj}(t) = \begin{cases} \Delta\theta_{obj} \oplus \theta_{obj}(0) & \text{if } t < t_m \\ \Delta\theta_{obj}\, e^{-(t-t_m)/\tau_\theta} \oplus \theta_{obj}(t) & \text{if } t \geq t_m \end{cases} \tag{14b}$$

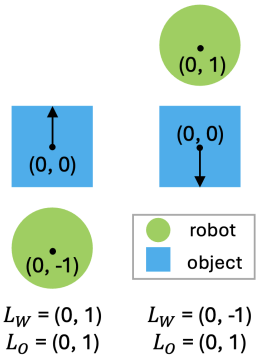

Fig. 8: The Laplacian coorinate should stay the same when the object rotates 180°.

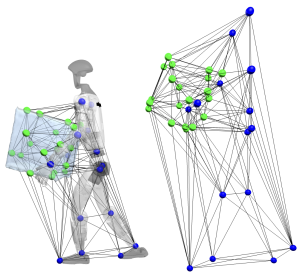

Fig. 9: The actual target (left) and source (right) interaction meshes used for optimization.

where τ_p and τ_θ are time constants governing the rate of decay for the translational and rotational perturbations, respectively. The $\oplus$ operator denotes composition for orientations (e.g., quaternion multiplication).

2) Interaction Mesh Construction in Object Frame: For robot-object interactions, it is crucial to construct the interaction mesh in the object's local coordinate frame. This ensures that the Laplacian coordinates, which encode relative spatial relationships, are invariant to the object's global rotation and translation. As illustrated in Fig. 8, when the object rotates by 180° (indicated by the black arrow), the Laplacian coordinate of the object in the world frame L_W changes from $(0,1)$ to $(0,-1)$, while the Laplacian coordinate calculated in the object frame L_O remains constant. Using object-frame coordinates is therefore essential for preserving the intended interaction geometry during object spatial transformation.

D. Sequential SOCP Details

At each time step t, we iteratively solve a Second-Order Cone Program (SOCP) for the optimal change in the robot's configuration dq util convergence for up to 10 iterations. For conciseness, we omit the time index t for variables within the Sequential SOCP loop.

The optimization finds the increment dq_n at the n-th iteration. The configuration is updated using this increment, starting from the previous time step's solution ($\bar{q}_0 = q^\star_{t-1}$):

$$\bar{q}_{n+1} = \bar{q}_n + dq^\star_n.$$

The optimal increment $dq^\star_n$ is the solution to the following SOCP, which is linearized around the current iterate $\bar{q}_n$:

$$dq^\star_n = \arg\min_{dq_n} \quad \|L^{\text{source}} - (J^n_L \cdot dq_n + \bar{L}^{\text{target}}_n)\|^2 \tag{15a}$$

$$+ \|\bar{q}_n + dq_n - q_{t-1}\|^2_Q \tag{15b}$$

$$\text{s.t.} \quad J^n_j \cdot dq_n + \phi_j(\bar{q}_n) \geq 0, \forall j \tag{15c}$$

$$q_{\min} \leq \bar{q}_n + dq_n \leq q_{\max} \tag{15d}$$

$$v_{\min} \cdot dt \leq \bar{q}_n + dq_n - q_{t-1} \leq v_{\max} \cdot dt \tag{15e}$$

$$p^F_t(\bar{q}_n) + J^n_F \cdot dq_n = p^F_{t-1}, \forall \text{stance foot} \tag{15f}$$

$$\|dq_n\|_2 \leq \varepsilon, \tag{15g}$$

where

- $L^{\text{source}} = vec(\{L(p^{\text{source}}_{t,i})\})$
- $L^{\text{target}}(q) = vec(\{L(p^{\text{target}}_{t,i}(q))\})$
- $\bar{L}^{\text{target}}_n = vec(\{L(p^{\text{target}}_{t,i}(\bar{q}_n))\})$
- $J^n_L = \partial L^{\text{target}} / \partial q|_{q=\bar{q}_n}$
- $J^n_j = \partial \phi_j / \partial q|_{q=\bar{q}_n}$
- $J^n_F = \partial p^F_t / \partial q|_{q=\bar{q}_n}$.

The second-order cone constraint (15g) is a trust region constraint with radius ε (we use $\varepsilon = 0.2$) that keeps the step size small, ensuring the linear approximations remain valid.

E. Downstream RL Evaluation Breakdown

Shown in Fig. 10, we present histograms from the downstream RL evaluation (Sec. V-B.0.b) to illustrate failure patterns and variance across OmniRetarget and baselines. These histograms break down failure rates by each motion for two tasks: robot–object interaction and robot–terrain interaction, highlighting not only overall averages but also how failures distribute across different motions. We do not include augmented motions as baselines do not support augmentation.

In robot–object interaction, the motions are modest while object properties are heavily randomized. Since the motions are not aggressive, most policies can adapt even to low-quality references and achieve at least one success, except VideoMimic, which fails systematically due to poor interaction preservation. This task therefore measures robustness rather than accuracy. We see that GMR shows broader failure spread with lower success rates, likely due to penetration issues that reduce robustness under placement changes. PHC shows improved robustness, while OmniRetarget achieves the most robust performance, with results concentrated in the high-success region.

In contrast, climbing terrains requires much more agile and challenging motions and thus, demands precise reference motions: if the quality is low, the agent fails outright with no successes. Here, PHC and VideoMimic perform the worst, with nearly half the motions failing entirely. GMR delivers somewhat better references but still fails on four motions, while OmniRetarget fails on only one. These results show that OmniRetarget not only provides superior robustness under variation but also higher reference accuracy.

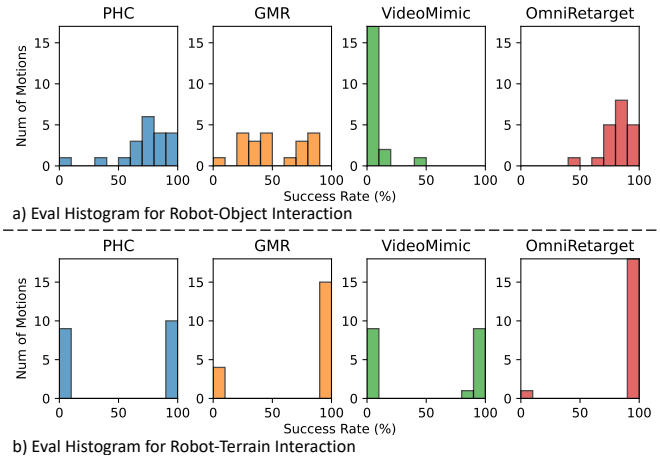

Fig. 10: Histograms from the downstream RL evaluation showing the failure patterns for the baselines in different tasks.

For the one remaining failure, we believe that it is limited by the simple RL formulation we use. For future work, an interesting direction could be to extend the current RL formulation with curriculum learning to support these extremely difficult motions.