OTRetarget: Joint Robot and Object Motion Retargeting via Optimal Transport

Guillaume Besset[*,1], Erwann Carn[*,1], Timothée Carecchio[1], Valentin Tordjman–Levavasseur[1], Fabian Schramm[1], Yann de Mont-Marin[1], Justin Carpentier[1], Ajay Suresha Sathya[1,2]

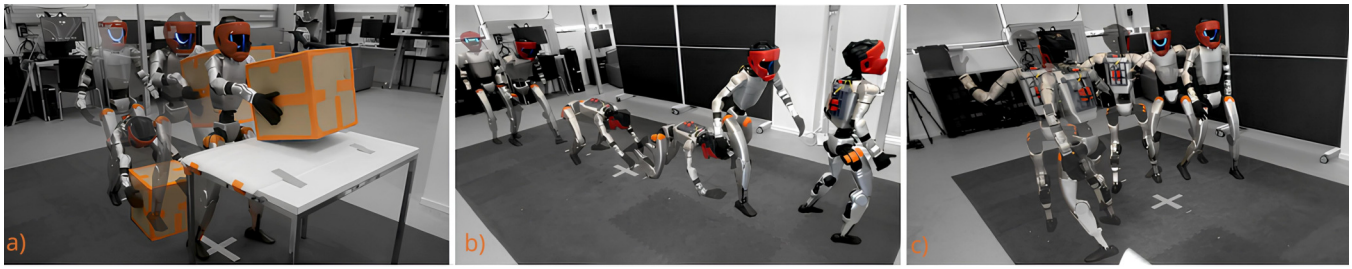

Fig. 1: **OTRETARGET for whole-body loco-manipulation.** Our approach jointly retargets robot and object motion while preserving interactions with the ground and surrounding objects, without rescaling the scene or demonstration. Overlaid poses show successive instants of the retargeted motions: **a)** two-handed box pickup and placement onto a table; **b)** quadrupedal crawling with hand–ground contact; and **c)** locomotion with a full-body spin.

***Abstract*— Transferring human motion to humanoid robots requires adapting the demonstrated motion to the robot morphology while preserving interactions with the environment. This is particularly challenging for loco-manipulation tasks, where contacts with the ground and manipulated objects must remain consistent despite differences in body proportions. Yet, skeletal motion alone does not fully describe these interactions, and fixing object trajectories limits the adaptation to a new embodiment. In this paper, we introduce OTRETARGET, a unified approach to jointly retarget robot and multi-object motion from human demonstrations. Our approach represents surface interactions through signed distances, closest surface points, and relative directions, and uses entropic optimal transport to transfer these quantities across human, robot, and object geometries. We incorporate the resulting interaction targets into a constrained inverse kinematics formulation that balances contact preservation with motion style and jointly optimizes robot and object poses at each frame. This formulation accommodates robot–object and object–object interactions without rescaling the scene or the demonstration. We validate the proposed approach on OMOMO, where it achieves a robot–object interaction Jaccard score of 87% and a depth error of 8.7 mm, compared with 28% and 29.3 mm for OmniRetarget. Finally, we demonstrate transfer to a physical G1 humanoid using whole-body policies trained with reinforcement learning on the retargeted references, across motions including two-handed box pick-and-place onto a table.**

I. INTRODUCTION

Imitating human motion has attracted growing interest in character animation and humanoid control [1], [2]. Commonly, a human motion is *retargeted* into a kinematic reference trajectory, a *whole-body tracking policy* learns to follow it [3], and the policy is deployed on the physical robot [4], [5], [6]. Because the policy learns from this retargeted reference, its embodiment gap-related artifacts propagate to the learned policy unless reward engineering compensates for them [7].

In free-space locomotion, the embodiment gap is largely one of *scale*: rescaling the source motion, done right, removes most of it [7], with reward shaping and domain randomization covering the rest [6]. Contact-rich motions, such as crawling or carrying a box, make retargeting errors more consequential. If a foot skates or a hand fails to make contact with the payload, learned behavior could fail regardless of joint tracking accuracy. In whole-body loco-manipulation [6], [8], the demonstrated interactions need to be preserved while adapting robot and object trajectories. This creates three related failure modes. **(i) Scaling inconsistency.** Rescaling shrinks the robot and object trajectories, not the scene, so contacts no longer align with the floor or table. **(ii) Unadapted object trajectory.** The demonstrated box trajectory lifted from the floor onto a table can be carried at heights unreachable for the robot when unscaled, and might deposit the box *under* the table when heights are rescaled. **(iii) Interaction loss.** Even for feasible trajectories, contact must adapt across embodiments: a shorter-armed robot might need to open its arms wider to appropriately grasp a box.

Existing retargeting methods leave at least one of these challenges unresolved: their reliance on skeletal motion provides only a partial description of the surface interactions that govern contact. To address these limitations, we introduce OTRETARGET, a proximity-aware method for whole-body loco-manipulation that jointly adapts robot and object motion using interaction targets extracted from scene geometry (Fig. 2). Our approach operates at the retargeting stage of the imitation pipeline, where human demonstrations are converted into robot reference trajectories.

Our approach represents interactions directly through the

[*]Equal contribution.
[1]Inria, Département d'Informatique de l'École Normale Supérieure, PSL Research University, Paris, France. firstname.lastname@inria.fr
[2]Dept. of Aeronautics and Astronautics, Stanford University, CA, USA.
Project page: simple-robotics.github.io/publications/otretarget

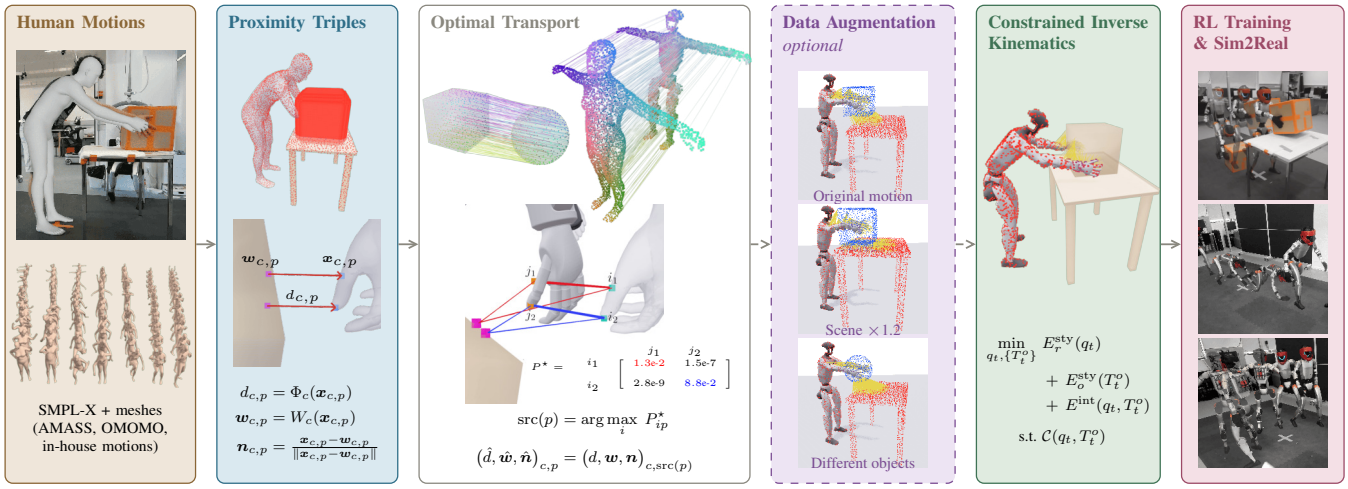

Fig. 2: **Pipeline overview.** Each probe $\boldsymbol{x}_{c,p}$ reads a proximity triple $(d, \boldsymbol{w}, \boldsymbol{n})_{c,p}$ from the channel's signed distance field. A transport plan $P^\star$, computed once, transfers this target to the robot. At each frame, a constrained program balances the resulting residuals against the skeleton-style targets while jointly solving the robot pose q_t and every object pose T_t^o. Dashed: the optional object substitution (Sec. III-E).

geometry of the participating surfaces. Signed distances describe contact and separation, while closest surface points and relative directions capture where and how these interactions occur. We evaluate these quantities at densely sampled surface points and use optimal transport to establish correspondences across human, robot, and object geometries. This representation lets us formulate retargeting as a balance between preserving the demonstrated interactions and retaining the motion style encoded by skeletal and object trajectories. Our work makes three contributions:

C1. Surface-level interaction residuals. We introduce pointwise interaction residuals extracted from demonstration geometry and integrate them with motion-style objectives in a per-frame constrained inverse kinematics problem. Our approach captures surface relationships that skeletal targets alone cannot express, addressing interaction loss across embodiments (iii).

C2. Joint robot and multi-object retargeting. Our formulation jointly optimizes robot and object poses, allowing their relative motion to adapt to the target embodiment while preserving robot–object, object–object, and ground interactions. By treating object trajectories as decision variables, our approach accommodates multiple interacting objects without rescaling the scene or demonstration, addressing scaling inconsistency (i) and unadapted object trajectories (ii).

C3. Interaction transfer across object geometries. We extend the optimal transport correspondence used for human-to-robot retargeting to map interaction targets between demonstrated and substitute objects. Combined with joint pose optimization, this enables our approach to adapt a single demonstration to previously unseen object shapes and sizes, generating multiple retargeted motion variants.

II. Related Work

In this section, we review motion retargeting and interaction modeling approaches relevant to whole-body loco-manipulation.

Point-stream retargeting maps human motion representation, such as joints [7], keypoints [2], or a learned latent [9], to robot poses. It includes kinematic adaptation to a new skeleton [10], learned models [11], [9] and general-purpose retargeters used in current humanoid pipelines [7], [2]. These methods optimize pose tracking and model neither the ground nor manipulated objects; thus, they address none of (i)–(iii).

Relational retargeting carries the *relative* configuration of body, object and terrain across embodiments. The original interaction-mesh method tetrahedralizes body joints and the scene vertices per frame, then minimizes the mesh deformation [12]. Relationship descriptors instead weight sampled surface points by proximity [13], interaction meshes were adapted to bipedal locomotion [14] and manipulation [15], and, through a spatial map between two static objects, onto a different object [16]. Throughout, the object pose is an input and not a free decision variable [15], leaving (i) and (ii) unsolved. OmniRetarget applies an interaction mesh to whole-body loco-manipulation, but solves only for the robot and globally rescales the demonstration by the robot-to-human height ratio [6]; a scene-scale extension keeps the same formulation [17]. Other methods do not model a rigid object at all [18], [19].

Limitations of interaction meshes. Existing relational methods carry interaction through sampled points. In a uniform-weight Laplacian [6], each keypoint couples to its mesh neighbors by connectivity [12], not by involvement in the manipulation. Interaction accuracy and posture are coupled in a single metric, and the optimization cost scales poorly with each sampled point. Sparse sampling improves tractability at the cost of local contact accuracy. Surface-level relations have been used for self-contact or a static partner in skinned characters [20], [21] but not for a humanoid interacting with a free object.

Dynamics-based recovery produces dynamically feasible motion through trajectory optimization before tracking [22], interaction-aware tracking policies [8], [23], a bilevel loop

around a tracking policy [24], or direct consumption of the demonstration [25]. They partly address (iii), while (i) and (ii) persist in methods that track a fixed object reference. These approaches focus on the dynamic feasibility of tracking a retargeted motion reference and complement our approach, which aims to preserve a demonstrated interaction.

Human–object interaction. We use two ideas from the human–object interaction literature: estimating the object pose jointly with the body [26] and representing the relation with continuous proximity [27]. These methods estimate or synthesize [28] interactions for an observed object, or imitate them in simulation [29]; our goal is to transfer a demonstration to another embodiment. None of the retargeting solvers above jointly decides the object trajectory and carries the demonstrated interaction as a continuous interaction residual.

III. Method

A. Problem setting and formulation

Conventionally, retargeting pipelines take a human demonstration, typically a video, as input and extract the skeleton and object pose trajectories. As discussed in Sec. I, approaches based on human skeleton poses use a coarse approximation of the human geometry, which is insufficient for accurately defining contact. Our approach instead densely samples points on human and object surfaces to obtain a more detailed geometry representation. We similarly sample points on the target robot and objects and *map* them to their corresponding references to specify desired interactions in the target scene. Deviations from the extracted skeletal and object pose trajectories, which encode the style of motion, and from the target interactions are used as cost terms in a constrained inverse kinematics (IK) problem solved frame by frame:

$$\begin{aligned} \min_{q_t,\ \{T_t^o\}} \quad & E^{\text{int}} + E_r^{\text{sty}} + E_o^{\text{sty}} + \left\| q_t \ominus q_{t-1} \right\|^2 \\ \text{s.t.} \quad & \mathcal{C}(q_t, \{T_t^o\}), \end{aligned} \tag{1}$$

where $q_t \in \mathcal{Q} \simeq \mathbb{R}^{n_q}$ is the robot configuration at time t, $\{T_t^o\} \in \mathrm{SE}(3)^{n_o}$ the poses of all manipulated objects. $E^{\text{int}}(q_t, \{T_t^o\})$ is the deviation cost for the target interaction (Sec. III-C), $E_r^{\text{sty}}(q_t)$ and $E_o^{\text{sty}}(\{T_t^o\})$ are the style deviation costs of the robot and objects, respectively (Sec. III-D). $\mathcal{C}$ encodes joint limits, joint velocity limits, per-object velocity limits and collision avoidance (including self-collision).

B. Surface-level proximity and optimal transport

To model surface-level interaction between two objects, we define proximity measures for points sampled on the objects' surfaces. Let $\boldsymbol{x}_p$, a point in the first object's surface point cloud indexed by p, denote a *probe* point. Let the second object, called *channel* c, have pose $T_t^c = (R_c, \boldsymbol{\tau}_c)$. The probe's position in the channel's frame is $\boldsymbol{x}_{c,p} = R_c^\top(\boldsymbol{x}_p - \boldsymbol{\tau}_c)$. We define the *proximity triple* for the probe w.r.t. the channel as:

$$\big(d, \boldsymbol{w}, \boldsymbol{n}\big)_{c,p} = \Big(\Phi_c(\boldsymbol{x}_{c,p}),\ W_c(\boldsymbol{x}_{c,p}),\ \tfrac{\boldsymbol{x}_{c,p} - \boldsymbol{w}_{c,p}}{\|\boldsymbol{x}_{c,p} - \boldsymbol{w}_{c,p}\|}\Big), \tag{2}$$

where $d_{c,p} = \Phi_c(\boldsymbol{x}_{c,p})$ is the probe's signed distance to the channel surface, $\boldsymbol{w}_{c,p} = W_c(\boldsymbol{x}_{c,p})$ the *witness* point, i.e. the closest point on the channel's surface, and $\boldsymbol{n}_{c,p}$ the unit vector pointing from witness to probe. The triple, defined in the channel frame, remains meaningful as the channel moves. Every entity, i.e. the human, the robot, the objects, carries a probe cloud. Φ_c and W_c are precomputed offline on a voxel grid around each object at 1 cm isotropic resolution and up to $\bar{d} = 15$ cm from the surface, then linearly interpolated to obtain the value. The grid is defined in object frame and hence invariant to channel pose transformation and does not require online recomputation. For a flat ground, Φ_c and W_c are affine. Probe clouds are sampled at 1000 pts/m^2 on objects and 2000 pts/m^2 on the human and the robot, and a probe–channel pair is active if the probe–channel distance is less than $\bar{d}$.

Optimal transport. Retargeting using proximity measures requires correspondences between source and target surfaces. Each human part is manually pre-assigned to a robot link. For each pair, we cast the correspondence problem between their surface point clouds as an entropically regularized optimal transport problem. We center and normalize point cloud coordinates by their root-mean-square radius to avoid sensitivity to size differences. We denote by a, b the uniform marginals derived from the respective objects' point cloud counts and by $C = (C_{ip})$ the matrix of squared Euclidean distances where i and p are the human part and robot link point indices, respectively. The resulting transport plan is obtained by solving

$$P^\star = \arg\min_{P \in \Pi(a,b)} \langle P, C\rangle + \varepsilon_{\text{ot}} \textstyle\sum_{i,p} P_{ip}\big(\log P_{ip} - 1\big), \tag{3}$$

where $\Pi(a,b) = \{P \geq 0 :\ P\mathbf{1} = a,\ P^\top\mathbf{1} = b\}$ is the transport polytope of couplings with marginals a and b, $\langle P, C\rangle := \mathrm{tr}(P^\top C)$ denotes the Frobenius inner product, and $\varepsilon_{\text{ot}} = 0.1$ is the entropic regularization parameter. We solve (3) using the Sinkhorn algorithm [30]. The correspondence for p is assigned as:

$$\mathrm{src}(p) = \arg\max_i\ P^\star_{ip}, \tag{4}$$

and the desired robot triples for any channel c follow from the demonstration triples, $(\hat{d}, \hat{\boldsymbol{w}}, \hat{\boldsymbol{n}})_{c,p} = (d, \boldsymbol{w}, \boldsymbol{n})_{c,\mathrm{src}(p)}$.

Unlike nearest-neighbor assignment, the marginal constraints in problem (3) force every human sample to receive mass, while the entropic term controls how sharply that mass concentrates. Considering top decile points in terms of distance from point cloud centroid, nearest-neighbor assignments leave 42% of these extremal samples unmatched against 14% for optimal transport. Coverage is thus more even, in particular over the limb extremities. The transport is pre-computed once per (human part, robot link) pair by setting both the robot and human in a T-pose, as shown in Fig. 2.

C. Interaction residuals

For any channel c and point p, interaction residuals between demonstration and target are defined using the proximity

triple:

$$
\begin{aligned}
r^d_{c,p} &= d_{c,p} - (\hat{d}_{c,p})_+, \\
r^x_{c,p} &= \rho_c\big(\boldsymbol{w}_{c,p}\,;\ \hat{\boldsymbol{w}}_{c,p}\big), \\
r^n_{c,p} &= \Big(\operatorname{sign}(\hat{d}_{c,p})\,\hat{\boldsymbol{n}}_{c,p}\cdot\big(\boldsymbol{x}_{c,p} - \hat{\boldsymbol{w}}_{c,p}\big)\Big)_-,
\end{aligned}
\tag{5}
$$

where $\rho_c(\boldsymbol{w};\hat{\boldsymbol{w}})$ is the geodesic distance on the channel surface, $(\cdot)_+ = \max(\cdot,0)$ and $(\cdot)_- = \min(\cdot,0)$. Geodesic distance is precomputed for each channel surface point cloud, and the distance to an arbitrary witness point is estimated by interpolation. The demonstrated distance is clamped, $(\hat{d}_{c,p})_+$, so that a noisy demonstration cannot impose a penetration as target. The residual r^n penalizes deviation from desired direction of interaction, and resolves the face ambiguities for cases when distance alone does not provide sufficient signal (the wrong side of a thin plate near the edge). The interaction cost E^{int} for the IK problem is defined as a weighted sum of quadratic residual losses:

$$
\begin{aligned}
E^{\text{int}} = \sum_{c,p} \frac{f_{\ell(p)}}{N_{\ell(p)}} \Big[\ & \lambda^d \beta^2_{c,p} \left(r^d_{c,p}\right)^2 \\
& + \lambda^x \bar{\beta}^2_{c,p} \left(r^x_{c,p}\right)^2 + \lambda^n \bar{\beta}^2_{c,p} \left(r^n_{c,p}\right)^2 \Big],
\end{aligned}
\tag{6}
$$

where $N_{\ell(p)}$ is the number of active probes on the link $\ell(p)$ carrying p, so that a link's authority does not grow with its sampled area, and $f_{\ell(p)}$ is a per-link weight, kept at 1 throughout. The weights are $(\lambda^d, \lambda^x, \lambda^n) = (256, 625, 2500)$ for probes carried by the robot, and four times those for probes carried by an object. The summation is over active channel-probe pairs (c,p) (Sec. III-B) in either the demonstration or in the retargeted scene: a pair that was close in the demonstration is pulled toward its demonstrated gap; a pair that is close now but was not close in the demonstration is instead pushed back toward that gap.

Both weights $\bar{\beta}, \beta$ are quadratic kernels of width $\sigma = 13\,\text{cm}$, so a pair's contribution is attenuated at distances beyond σ. The first, $\bar{\beta} = \big(1 - (\hat{d})_+/\sigma\big)^2_+ \le 1$, depends on the demonstration distance $\hat{d}$ alone, so that r^x and r^n, which determine tangential motion, are active only when the demonstration exhibits contact. The second, $\beta = \min\big((1 - \min\big((\hat{d})_+, d\big)/\sigma\big)^2_+, \beta_{\max}\big)$, uses the smaller of demonstrated $\hat{d}$ and the current distance d, to control distance when either demonstration or the retarget scene has an active contact pair. Since d is signed, penetration ($d < 0$) can drive the kernel above 1, and is capped at $\beta_{\max} = 5$.

D. Joint kinematic retargeting solver

Style tracking cost. The style cost uses the human skeleton poses as a motion reference through postural costs. Let $h(\ell)$ map a robot link ℓ to the corresponding human link. For the L tracked robot links, the orientation cost penalizes deviations from the desired orientation reference $\hat{R}_\ell = R_{h(\ell)} O_\ell$, where the world orientation of the human skeleton $R_{h(\ell)}$ is corrected by a fixed alignment offset O_ℓ. We align the robot and human pelvises at $\hat{x}_0$ with orientation $\hat{R}_\ell$, then compute target link positions by computing forward kinematics over the kinematic tree using the desired orientations and the robot's own link lengths. Deriving positions from desired orientations avoids the conflicts between the two that scaling-based methods incur when deforming the human skeleton to the robot's morphology. The link length ratios can range from 0.19 to 1.18, with no scaling factor consistent across links. Tracking the full skeleton avoids overweighting the position of any single link. The style costs for links and object frames are

$$
\begin{aligned}
E^{\text{sty}}_r &= \sum_{\ell} \Big[\lambda^{\text{pos}} \big\| x_\ell(q) - \hat{x}_\ell \big\|^2 + \lambda^{\text{rot}} \big\| R_\ell(q) \ominus \hat{R}_\ell \big\|^2 \Big], \\
E^{\text{sty}}_o &= \sum_{o} \Big[\lambda^{\text{pos}}_o \big\| \tau^o - \hat{\tau}^o \big\|^2 + \lambda^{\text{rot}}_o \big\| R^o \ominus \hat{R}^o \big\|^2 \Big],
\end{aligned}
\tag{7}
$$

with $R \ominus \hat{R} = \log(R\hat{R}^\top)^\vee$ the world-frame orientation error and λ weighting position against orientation. The style costs carry no robot-object interaction term; the object term centers each manipulated object to its demonstrated trajectory.

Native scale initialization. The robot's root pose and the scene are retained at the captured metric scale, and the target link positions are computed using forward kinematics. To permit link positions to be determined predominantly by the demonstrated interaction, feet–ground contact included, the position weight λ^{pos} is kept small. The motion of a human picking an object off a table at a given height must retarget to a robot reaching that same height, regardless of the scale difference; the position targets nevertheless provide a posture prior for links not involved in interaction.

Hard kinematic constraints. Collision avoidance between the robot, the objects and the ground also uses the signed distance fields as a hard non-penetration constraint $d_{c,p} \ge -\varepsilon_{\text{col}}$ with $\varepsilon_{\text{col}} = 0.3$ mm on the sampled points of every probe–channel pair. Self-collision uses capsules approximating the robot links and applies the same criterion. Each collision pair contributes an inequality constraint in the optimization problem. In addition to the usual robot joint position and velocity limits, for each object i, the boxes $\mathcal{B}_{\text{env}}$ and $\mathcal{B}_{\text{cor}}$ of (8e) denote feasible sets that constrain displacement between consecutive frames and deviation from reference motion, respectively: $|\delta\xi_i + \xi^{\text{acc}}_i| \le b^{\text{env}}_i$ and $|\delta\xi_i + \xi^{\text{cum}}_i| \le b^{\text{cor}}_i$, where ξ^{acc}_i is the accumulated single time-step displacement and ξ^{cum}_i the cumulative one since the phase origin. Fixed objects can be modeled via $b^{\text{cor}} = 0$, or, better, by not making their pose a free decision variable in the optimization problem.

Sequential optimization solver. We solve (1) using a trust-region sequential quadratic programming (SQP) approach. At each SQP iterate, we employ a Gauss–Newton Hessian approximation and linearize all constraints to obtain a quadratic program (QP) as the inner problem. The QP's decision variables are step changes between SQP iterations in robot configuration δv (floating base and joints) and object pose $\delta\xi_o = (\delta\tau_o, \delta\theta_o) \in \mathbb{R}^6$ per manipulated object, retracted as $\tau^o \leftarrow \tau^o + \delta\tau_o$, $R^o \leftarrow \exp(\delta\theta_o)\, R^o$. Every manipulated

object is a decision variable. The QP subproblem is:

$$\min_{\delta v,\, \delta\xi} \quad \widehat{E}_r^{\mathrm{sty}} + \widehat{E}_o^{\mathrm{sty}} + \widehat{E}^{\mathrm{int}} + \left\|(q_t \oplus \delta v) \ominus q_{t-1}\right\|^2 \tag{8a}$$

$$\text{s.t.} \quad q^{\min} \le q_t \oplus \delta v \le q^{\max}, \tag{8b}$$

$$\left|(q_J \oplus S\delta v) \ominus q_{J,t-1}\right| \le v_{\max}\,\Delta t, \tag{8c}$$

$$d_{c,p} + \boldsymbol{g}_{c,p}^{\top}(\delta v, \delta\xi) \;\ge\; -\varepsilon_{\mathrm{col}}, \tag{8d}$$

$$\delta\xi \in \mathcal{B}_{\mathrm{env}} \cap \mathcal{B}_{\mathrm{cor}}, \tag{8e}$$

$$|\delta v| \le \Delta_v, \quad |\delta\xi| \le \Delta_\xi, \tag{8f}$$

where $\widehat{E}_r^{\mathrm{sty}}, \widehat{E}_o^{\mathrm{sty}}, \widehat{E}^{\mathrm{int}}$ are the Gauss–Newton quadratic objective approximations, q_J the actuated joint angles at the iterate, S their selector in δv, $v_{\max}$ the joint velocity limits, and Δ_v, Δ_ξ fixed per-DoF boxes (5 cm, 0.10 rad). The loop runs a fixed budget of 50 iterations on the cold-started first frame and six warm-started thereafter, or stops when the step falls below 10^{-4}; the QP is assembled from the analytic frame Jacobians of Pinocchio [31] and solved with ProxQP [32], warm-started across SQP iterations and frames.

E. Generalization to new object shapes

Our approach extends to new object shapes and sizes using a single demonstration. We hypothesize that preserving the approach motion and contact relationships enables interaction transfer between objects with similar surface topology. To this end, we reuse the optimal transport formulation introduced for human-to-robot correspondence in Sec. III-B to map interaction targets from the demonstrated object to a substitute object.

We precompute correspondences between the two surface point clouds, each expressed in its object's local frame and normalized to unit radius. This normalization is used only to establish correspondences: witness locations are mapped onto the substitute surface, while the interaction targets retain their metric meaning without rescaling.

The mapped targets are then incorporated into the joint retargeting problem, with the substitute object's pose as a decision variable. This allows robot–object and object–object relative motions to adapt to the new geometry while preserving the demonstrated interactions.

IV. Experiments

We evaluate OTRetarget on motion and interaction fidelity, runtime, object generalization, and hardware transfer through learned tracking policies.

A. Experimental setup and evaluation protocol

Robot, demonstrations, and baselines. All demonstrations are retargeted to the 29-DoF Unitree G1 with fixed hands, for which a *grip* is defined by hand–object proximity. Our evaluation covers all 2 027 body-only motions from the CMU and SFU subsets of AMASS [33], all 4 421 single-object manipulation sequences from 13 OMOMO categories [28] in the InterMimic release [29], and our own two-object RGB-D sequence (*pnp14*). For the latter, we recover human motion with GVHMR [34] and object poses with FoundationPose [35], using primitive meshes at measured dimensions for object geometry. We compare against OmniRetarget [6], GMR [7], and PHC [2] under their released settings, marking any adaptations. Since GMR and PHC do not model objects, they are excluded from OMOMO. OmniRetarget reports infeasibility on 424 OMOMO sequences, mostly involving chairs and large tables. Omitting these failures from its aggregate metrics slightly favors the baseline.

Each method retains its scaling convention: OmniRetarget and our scaled ablation scale human and object trajectories by $sc =$ robot height/stature without resizing objects, where G1 is 1.32 m tall and stature denotes the SMPL-X rest-mesh height. GMR uses its released root-path scale, $sc = 0.9 \times$ stature/1.8, while PHC and our native-scale variants use $sc = 1$. We report the median sc for each dataset.

Interaction metrics. We represent each interaction by a sample (t, u, c) pairing a body unit u with a channel c (ground or object) at frame t, and record its signed distance and closest surface point in the demonstration, $(\hat{d}, \hat{\boldsymbol{w}})$, and on the robot, $(d, \boldsymbol{w})$. For all methods, these quantities are computed independently of the solver using exact signed distances from posed SMPL-X and robot visual-mesh vertices to the channel mesh, with each unit's distance determined by its closest vertex.

With a threshold of $\varepsilon_{\mathrm{eval}} = 20$ mm, the demonstrated and retargeted interaction sets are $\mathcal{H} = \{(t,u,c) \,|\, (\hat{d})_+ < \varepsilon_{\mathrm{eval}}\}$ and $\mathcal{R} = \{(t,u,c) \,|\, (d)_+ < \varepsilon_{\mathrm{eval}}\}$. We measure their agreement through precision $P = |\mathcal{H} \cap \mathcal{R}|/|\mathcal{R}|$, recall $R = |\mathcal{H} \cap \mathcal{R}|/|\mathcal{H}|$ and Jaccard index $J = |\mathcal{H} \cap \mathcal{R}|/|\mathcal{H} \cup \mathcal{R}|$. To assess geometric accuracy over $\mathcal{H}$, we also report depth error $dd = \mathrm{mean}\,|d - (\hat{d})_+|$ and placement error $dw = \mathrm{mean}\,\|\boldsymbol{w} - \hat{\boldsymbol{w}}\|$, measuring deviations in gap and surface location, respectively. Ground and object interactions are reported separately; penetrations count as interactions, with their magnitude captured by dd on $\mathcal{H}$.

Additional metrics. We assess motion style using *rot*, the mean link-orientation error in degrees relative to the style target, with the same SMPL-X-to-robot alignment O_ℓ for all methods. Position error is omitted because leg-length differences dominate it. For self-collision, *self* measures the 95th-percentile penetration depth from the robot visual mesh, independently of the solver's capsules. Foot contact is characterized by *skate*, the median sliding velocity of a foot's least-moving point during stance, and *clear*, its median stance height. Finally, *drift* measures mean object-position deviation from the demonstration and is zero for fixed trajectories. Our objective does not directly optimize *self*, *skate*, or *clear*.

Aggregation and parameter settings. Within each sequence, we compute P, R, and J as sample ratios, average dd and dw over $\mathcal{H}$, and average *rot* and *drift* over frames. We then report dataset medians over the sequences returned by each method. OTRetarget uses the same weights across datasets and scenes, without per-sequence tuning. Baselines retain their released settings, except for the object-interaction weights in our OmniRetarget extension.

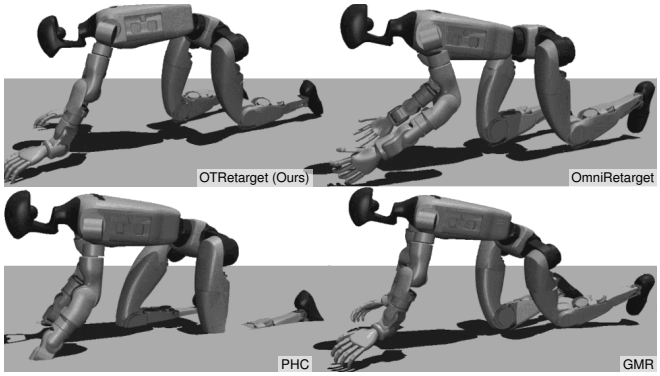

Fig. 3: **Hands and knees on the floor** (*crawl*). OTRETARGET lays both hands flat on the floor as the demonstration does. OmniRetarget reaches the floor with the wrists bent; GMR, which represents no terrain, leaves the hands above it, and PHC drives them through it.

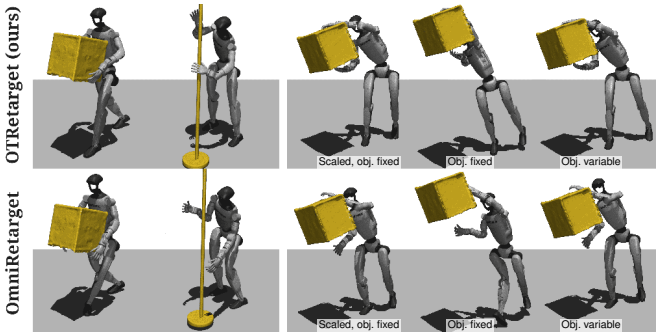

Fig. 4: **Illustration of loco-manipulation on OMOMO,** OTRETARGET (top) against OmniRetarget (bottom): a large convex box, then a floor lamp, a stem the mesh has no volume to hold (left); the same overhead lift in the scaled world, native with the object pinned, and native with it free (right).

B. Comparison with existing retargeting methods

Ground interaction. Table I evaluates every method over 2 027 AMASS and 4 421 OMOMO sequences. OTRETARGET achieves the best style error, floor depth, and interaction agreement on both datasets (Fig. 3). The *rot* gap to OmniRetarget and PHC is structural rather than a matter of tuning: both optimize joint positions without directly constraining link orientation, which causes the wrists to remain bent on the floor (Fig. 3) and the feet splayed outward during carrying (Fig. 4). GMR has the lowest *skate*, but its soles never settle on the floor (*clear*); OmniRetarget has the opposite failure with negative clearance. On OMOMO, the two interaction-aware methods retain ground contact while their style error increases because the robot must adapt its grip. OmniRetarget reaches 13.7 mm self-penetration on AMASS because self-collision is disabled by default; GMR and PHC constrain neither self- nor scene-collision.

Robot-object interaction. Table III reports robot–object interaction on OMOMO. OTRETARGET reaches 87% Jaccard agreement, versus 28% for OmniRetarget, with one third of its depth and placement errors. The scaled, fixed-object ablation retains most of this gain, showing that the interaction residuals, rather than the native scene or variable object, drive the improvement.

On a large convex box, OmniRetarget matches the demonstrated depth more closely but also predicts contact in non-contact frames, lowering precision (Fig. 4, left). On a floor lamp, whose stem affords a small grasp volume, it loses the interaction entirely; our result is unchanged. Thus, OmniRetarget's interaction mesh appears to be more sensitive to object geometry than OTRETARGET's surface stream.

Runtime. Our unoptimized Python implementation achieves median runtimes of 21.5 ms per frame for locomotion (*spin, CMU, 88_10*) and 63.8 ms for object carrying (*largebox, OMOMO, sub3_003*), measured on a single core of an AMD Threadripper PRO 7955WX. On these sequences, our approach is 12–22× faster than OmniRetarget, which requires 470 and 749 ms per frame, respectively. GMR remains faster at 3.2 and 2.5 ms, but does not model interactions with the environment. These results motivate further implementation optimization toward online retargeting.

C. Retargeting the native scene

Table II reports results for retargeting in a captured, native-scale scene and the benefits of a variable object. On OMOMO, fixing the object slightly worsens style and ground contact; freeing it recovers both with less than 3 cm drift. The median hides the important overhead-lift case (Fig. 4, right): a fixed

TABLE I: **Ground fidelity** on locomotion (AMASS, CMU and SFU pooled) and loco-manipulation (OMOMO), each method in its released world; dataset medians, sequence counts in the block headers.

Method	*sc*	rot↓ °	self↓ mm	Terrain dd↓ mm	R↑ %	P↑ %	J↑ %	skate↓ cm/s	clear mm
Locomotion - AMASS (2027 sequences)									
OTRETARGET	1.00	**6.3**	**0.0**	**1.2**	**99**	100	**99**	1.8	1.3
OmniRetarget	0.76	25.5	13.7	3.3	95	100	93	2.1	5.1
GMR	0.87	7.5	**0.0**	23.2	47	100	47	**1.3**	30.9
PHC	1.00	21.7	**0.0**	17.3	68	100	68	4.9	29.7
One-Object Manipulation - OMOMO (4421 sequences)									
OTRETARGET	1.00	**10.7**	**0.0**	**1.7**	**99**	100	**99**	3.5	4.6
OmniRetarget	0.75	27.8	2.0	2.7	98	100	96	**3.0**	-0.7

TABLE II: **The object as a decision variable.** The scene solved as captured (ablation: object fixed vs. variable); *drift* is blank on scaled rows, where it would measure the world's scale, not the method.

Method	*sc*	rot↓ °	Terrain R↑ %	Object R↑ %	P↑ %	J↑ %	drift cm
OMOMO (4421 sequences)							
OTRETARGET (scaled, obj. fixed)	0.75	11.1	**100**	**97**	89	82	—
native scene, obj. fixed	1.00	12.0	98	94	96	84	0.0
native scene, obj. variable	1.00	**10.7**	99	95	96	**87**	2.9
largebox, overhead lift (OMOMO, sub8_028)							
OTRETARGET (scaled, obj. fixed)	0.72	**8.8**	**100**	72	99	72	—
native scene, obj. fixed	1.00	16.8	79	71	98	71	0.0
native scene, obj. variable	1.00	9.8	99	**80**	99	**79**	12.2
OmniRetarget	0.72	27.0	99	48	82	44	—
native scene (our ext.)	1.00	34.7	48	46	79	41	0.0
+ ground in the graph (our ext.)	1.00	48.8	74	46	62	36	0.0
+ obj. variable (our ext.)	1.00	28.8	**100**	54	71	44	32.4

TABLE III: **Robot–object proximity** on the OMOMO dataset (medians) and on two single captures, a large convex box and a floor lamp; the marked row solves in OmniRetarget's scaled world with the object fixed.

Method	*sc*	Object dd↓ mm	dw↓ mm	R↑ %	P↑ %	J↑ %
OMOMO (4421 sequences)						
OTRETARGET	1.00	**8.7**	**37**	95	96	**87**
OTRETARGET (scaled, obj. fixed)	0.75	8.8	38	**97**	89	82
OmniRetarget	0.75	29.3	102	39	62	28
largebox (OMOMO, sub3_003)						
OTRETARGET	1.00	10.0	**42**	94	89	**85**
OTRETARGET (scaled, obj. fixed)	0.68	9.1	44	**97**	81	79
OmniRetarget	0.68	**5.7**	54	95	58	56
floorlamp (OMOMO, sub10_031)						
OTRETARGET	1.00	**9.3**	**37**	**99**	100	**99**
OTRETARGET (scaled, obj. fixed)	0.80	9.4	38	**99**	100	**99**
OmniRetarget	0.80	108.6	302	0	—	0

native object doubles the style error and lifts the feet, whereas a variable object moves 12 cm on average to a reachable height and recovers posture, stance, and grip. Our OmniRetarget extension with a variable object shows a similar trend. Adding ground to its graph improves ground contact but worsens posture; freeing the object improves both at the cost of almost three times our drift. But object grip remains poor, indicating the importance of the interaction term.

Multi-object retargeting. In *pnp14* (Fig. 1a and Fig. 2) a box is lifted from the floor onto a table; the two objects must make appropriate contact when the box comes to rest. Box-table and box-arm interactions' recall and precision remain high at 95% and 98%. The table object remains on the floor and the robot does not make spurious table contact. In flight, the box departs from its demonstrated trajectory by up to 27 cm to remain reachable, then settles on the table at the demonstrated place during demonstrated frames, with a 2.7 cm mean drift and an 8.8° *rot* error.

D. Retargeting generalization to object variations

Object resizing. Fig. 5 resizes the scanned box from ×0.7 to ×1.6. OTRETARGET's grip recall remains nearly constant because the proximity triples evaluated on the substituted surface and variable object enable contact at desired time instances. OmniRetarget is accurate only at its native size: recall collapses on smaller boxes and halves with OmniRetarget's Laplacian metric not sufficiently encoding desired contact as its mesh vertices are scaled. Our precision decreases for larger boxes because contact is made for a longer period, induced by style targets.

Object swapping. Table IV replaces the scanned box with eight unseen shapes, from a ball to a beam. The demonstration remains unchanged, while the object↔object transport plan of Sec. III-E maps each demonstrated witness to the substitute. Recall and precision remain near those of the native box and the depth error stays within 3 mm. Drift increases for shapes less similar to a box, reaching roughly twice the native value for the torus and board. This drift is the adaptation required to preserve the interaction, not an error that a fixed trajectory could remove.

TABLE IV: **One capture, multiple unseen objects:** the capture never touched (*largebox, OMOMO, sub3_003*), every row is obtained with OTRETARGET

Variant	Object dd↓ mm	R↑ %	P↑ %	J↑ %	drift cm
box ×1.0 (native)	10.0	94	89	85	9.9
ball ∅0.34	12.1	97	87	84	12.4
drum ∅0.34×0.36	12.2	95	88	84	11.1
capsule ∅0.23×0.36	11.8	96	88	84	13.1
rugby ball 0.43×0.31×0.28	9.4	97	86	84	13.8
torus ∅0.36	10.2	97	90	87	18.6
cube 0.26	9.9	89	90	81	13.3
board 0.38×0.30×0.12	10.1	96	88	85	16.8
beam 0.46×0.19×0.19	12.0	94	87	82	14.7

E. Policy training and hardware transfer

Policy training setup. The reinforcement learning policy is used only for downstream evaluation of reference quality and does not constitute a contribution. We use the Holosoma framework [6] and its published PPO [36] recipe, which uses 4096 environments, 30 000 iterations. An episode succeeds when it completes the clip without a tracking termination (a tracked body/torso deviates from its reference by more than 25/50 cm). Evaluations use the mean action.

Locomotion. Without an object, the original framework is used. We retarget ten AMASS clips, from crawling to spinning on one leg, with both methods and train a policy per clip under identical settings. Success rates are similar (98.8% vs. 97.2%), but tracking error differs significantly. Rolling out every saved checkpoint against its own reference, we measure mean per-joint position error $\mathrm{MPJPE} = \frac{1}{|\mathcal{B}|}\sum_{b\in\mathcal{B}}\left\|(\boldsymbol{p}_b - \boldsymbol{p}_{\mathrm{root}}) - (\hat{\boldsymbol{p}}_b - \hat{\boldsymbol{p}}_{\mathrm{root}})\right\|$ over the $|\mathcal{B}| = 14$ tracked bodies, simulated $\boldsymbol{p}$ against reference $\hat{\boldsymbol{p}}$, each relative to its own root so it reads posture, not path. Policies trained on our reference reach 28.8 mm versus 33.7 mm for OmniRetarget and lead by ≈ 5 mm throughout training (Fig. 6). The improvement is not due to an easier reference, as ours is closer to the demonstration (Tab. I). Using OmniRetarget's final MPJPE as a threshold, our policies reach it by iteration 6k, at one fifth of the baseline budget.

Manipulation. For manipulation, we demonstrate transfer rather than compare references. Holosoma's released object

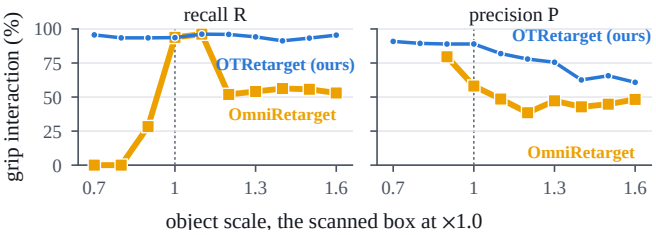

Fig. 5: **Illustration of one demonstration replayed across a distribution of object sizes** (*largebox, OMOMO, sub3_003*): grip recall and precision against the scale of the box.

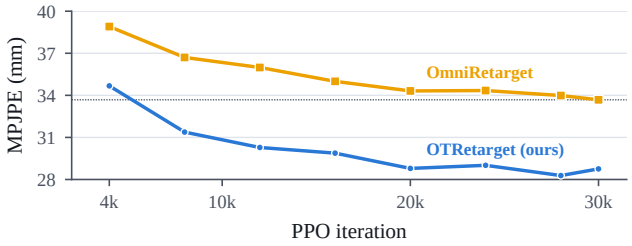

Fig. 6: **Impact of Retargeting Quality on Tracking Performance.** MPJPE against PPO iterations for ten AMASS clips using identical training pipelines. The dotted line marks OmniRetarget's final accuracy.

framework plateaus at 2% success on *pnp14* with our native, variable-object reference. Training requires object reference-aware addition to Holosoma, making an A/B comparison with the Holosoma framework misleading. We add box linear- and angular-velocity tracking and the force-set point reward of [8], gated by robot–box and box–table proximity windows from the reference. Training reaches 48% success under a 25 cm object termination, with the box 11 cm from its reference. The same policy runs on the physical G1 without retraining or object perception and completes 6 of 8 pick-and-place trials (supplementary video). This is a single-robot, single-clip demonstration rather than a hardware study.

V. Conclusion

We have introduced OTRetarget, a geometry-aware approach to jointly retarget human and multi-object motion to humanoid robots. Our formulation combines surface-level interaction targets with optimal transport to adapt robot and object trajectories while preserving contacts without rescaling the scene. Across two datasets comprising $6\,448$ clips, our approach improves both motion-style tracking and interaction fidelity over existing methods and supports transfer to unseen object shapes and sizes. Whole-body policies trained on the resulting references transfer to a physical G1 humanoid, including for two-handed box pick-and-place.

Our evaluation remains limited to one robot, flat ground, and a single two-object capture. The formulation is kinematic and operates frame by frame, without explicitly modeling forces or optimizing over a time horizon. Future work will explore more complex terrains and scenes, integration with dynamics-aware trajectory optimization, and online tracking policy synthesis to extend the approach across embodiments and environments.

Acknowledgments

This work was supported by the European Union's Horizon Europe research and innovation programme through the Marie Skłodowska-Curie Postdoctoral Fellowship ExTRAORDiNary (grant no. 101211945) and the ARTIFACT project (grant no. 101165695). Additional support was provided by the French government through the "PR[AI]RIE-PSAI" AI Cluster (ANR-23-IACL-0008), managed by the Agence Nationale de la Recherche, and through the France 2030 Organic Robotics Program (PEPR O2R) and the PIQ program, the latter managed by the Agence de Programme du Numérique. Views and opinions expressed are those of the author(s) only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the European Commission can be held responsible for them.

References

[1] X. B. Peng, P. Abbeel, S. Levine, and M. van de Panne, "DeepMimic: Example-guided deep reinforcement learning of physics-based character skills," *ACM Trans. Graph. (Proc. SIGGRAPH)*, vol. 37, no. 4, pp. 143:1–143:14, 2018.

[2] Z. Luo, J. Cao, A. W. Winkler, K. Kitani, and W. Xu, "Perpetual humanoid control for real-time simulated avatars," in *Proc. IEEE/CVF Int. Conf. Computer Vision (ICCV)*, 2023, pp. 10 895–10 904.

[3] 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," *Science Robotics*, vol. 11, no. 117, p. eadx8924, 2026.

[4] T. He, Z. Luo, W. Xiao, C. Zhang, K. Kitani, C. Liu, and G. Shi, "Learning human-to-humanoid real-time whole-body teleoperation (H2O)," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2024, pp. 8944–8951.

[5] 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. Fan, Y. Zhu, C. Liu, and G. Shi, "ASAP: Aligning simulation and real-world physics for learning agile humanoid whole-body skills," in *Proc. Robotics: Science and Systems (RSS)*, 2025.

[6] L. Yang, X. Huang, Z. Wu, A. Kanazawa, P. Abbeel, C. Sferrazza, C. K. Liu, R. Duan, and G. Shi, "OmniRetarget: Interaction-preserving data generation for humanoid whole-body loco-manipulation and scene interaction," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2026, arXiv:2509.26633.

[7] J. P. Araújo, Y. Ze, P. Xu, J. Wu, and C. K. Liu, "Retargeting matters: General motion retargeting for humanoid motion tracking," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2026, arXiv:2510.02252.

[8] H. Weng, Y. Li, N. Sobanbabu, Z. Wang, Z. Luo, T. He, D. Ramanan, and G. Shi, "HDMI: Learning interactive humanoid whole-body control from human videos," *preprint arXiv:2509.16757*, 2025.

[9] Q. Zhao, K. Yang, X. Wang, S. Zhao, Y. Lu, X. Zhang, W. Yin, Q. Shen, X.-X. Long, and X. Cao, "Make tracking easy: Neural motion retargeting for humanoid whole-body control," *preprint arXiv:2603.22201*, 2026.

[10] M. Gleicher, "Retargetting motion to new characters," in *Proc. SIGGRAPH*, 1998, pp. 33–42.

[11] K. Aberman, P. Li, D. Lischinski, O. Sorkine-Hornung, D. Cohen-Or, and B. Chen, "Skeleton-aware networks for deep motion retargeting," *ACM Trans. Graph. (SIGGRAPH)*, vol. 39, no. 4, pp. 62:1–62:14, 2020.

[12] E. S. L. Ho, T. Komura, and C.-L. Tai, "Spatial relationship preserving character motion adaptation," *ACM Trans. Graph. (Proc. SIGGRAPH)*, vol. 29, no. 4, pp. 33:1–33:8, 2010.

[13] R. A. Al-Asqhar, T. Komura, and M. G. Choi, "Relationship descriptors for interactive motion adaptation," in *Proc. ACM SIGGRAPH/Eurographics Symp. Computer Animation*, 2013, pp. 45–53.

[14] S. Nakaoka and T. Komura, "Interaction mesh based motion adaptation for biped humanoid robots," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots (Humanoids)*, 2012, pp. 625–631.

[15] J. Wu, S. Yao, G. He, X. Liu, Z. Zeng, X. Jiang, H. Yang, W. Zhang, and H. Zhao, "TopoRetarget: Interaction-preserving retargeting for dexterous manipulation," *preprint arXiv:2606.16272*, 2026.

[16] Y. Kim, H. Park, S. Bang, and S.-H. Lee, "Retargeting human-object interaction to virtual avatars," *IEEE Trans. Vis. Comput. Graph.*, vol. 22, no. 11, pp. 2405–2412, 2016.

[17] Q. Zhang, J. Ma, P. Liu, S. Shi, Z. Su, Z. Wang, J. Sun, W. Cui, J. Yu, G. Han *et al.*, "MeshMimic: Geometry-aware humanoid motion learning through 3D scene reconstruction," *preprint arXiv:2602.15733*, 2026.

[18] T. Cheynel, T. Rossi, B. Bellot-Gurlet, D. Rohmer, and M.-P. Cani, "ReConForM: Real-time contact-aware motion retargeting for more diverse character morphologies," *Comput. Graph. Forum*, vol. 44, 2025.

[19] W.-J. Huang, Y.-Y. Zhang, Y.-L. Wei, Z.-W. Xia, J. Tan, Y.-M. Li, Z. Zhao, and W.-S. Zheng, "Beyond mimicry: Learning whole-body human-humanoid interaction from human-human demonstrations," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, 2026, pp. 30 740–30 749, arXiv:2601.09518.

[20] T. Jin, M. Kim, and S.-H. Lee, "Aura mesh: Motion retargeting to preserve the spatial relationships between skinned characters," *Comput. Graph. Forum (Proc. Eurographics)*, vol. 37, no. 2, pp. 311–320, 2018.

[21] R. Villegas, D. Ceylan, A. Hertzmann, J. Yang, and J. Saito, "Contact-aware retargeting of skinned motion," in *Proc. IEEE/CVF Int. Conf. Computer Vision (ICCV)*, 2021, pp. 9700–9709.

[22] V. Dhédin, I. Taouil, S. Omar, D. Yu, K. Tao, A. Dai, and M. Khadiv, "DynaRetarget: Dynamically-feasible retargeting using sampling-based trajectory optimization," *IEEE Robot. Autom. Lett.*, 2026, arXiv:2602.06827.

[23] 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," *preprint arXiv:2510.05070*, 2025.

[24] D. Müller, A. Serifi, S. Christen, R. Grandia, E. Knoop, and M. Bächer, "ReActor: Reinforcement learning for physics-aware motion retargeting," *ACM Trans. Graph. (SIGGRAPH)*, vol. 45, no. 4, pp. 97:1–97:11, 2026.

[25] C. Roux, L. De Matteïs, A. Jordana, V. Guillet, N. Mansard, O. Stasse, and P. Souères, "Direct dynamic retargeting for humanoid imitation learning from videos," *preprint arXiv:2605.23762*, 2026.

[26] B. L. Bhatnagar, X. Xie, I. A. Petrov, C. Sminchisescu, C. Theobalt, and G. Pons-Moll, "BEHAVE: Dataset and method for tracking human object interactions," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 15 914–15 925.

[27] M. Hassan, V. Choutas, D. Tzionas, and M. J. Black, "Resolving 3D human pose ambiguities with 3D scene constraints (PROX)," in *Proc. IEEE/CVF Int. Conf. Computer Vision (ICCV)*, 2019, pp. 2282–2292.

[28] J. Li, J. Wu, and C. K. Liu, "Object motion guided human motion synthesis," *ACM Trans. Graph. (Proc. SIGGRAPH Asia)*, vol. 42, no. 6, pp. 197:1–197:11, 2023.

[29] S. Xu, H. Y. Ling, Y.-X. Wang, and L.-Y. Gui, "InterMimic: Towards universal whole-body control for physics-based human-object interactions," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, 2025, pp. 12 266–12 277.

[30] M. Cuturi, "Sinkhorn distances: Lightspeed computation of optimal transport," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 26, 2013.

[31] J. Carpentier, G. Saurel, G. Buondonno, J. Mirabel, F. Lamiraux, O. Stasse, and N. Mansard, "The Pinocchio C++ library – a fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives," in *IEEE International Symposium on System Integrations (SII)*, 2019.

[32] A. Bambade, S. El-Kazdadi, A. Taylor, and J. Carpentier, "PROX-QP: Yet another quadratic programming solver for robotics and beyond," in *Proc. Robotics: Science and Systems (RSS)*, 2022.

[33] N. Mahmood, N. Ghorbani, N. F. Troje, G. Pons-Moll, and M. J. Black, "AMASS: Archive of motion capture as surface shapes," in *Proc. IEEE/CVF Int. Conf. Computer Vision (ICCV)*, 2019, pp. 5442–5451.

[34] 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 *Proc. SIGGRAPH Asia Conf. Papers*, 2024.

[35] B. Wen, W. Yang, J. Kautz, and S. Birchfield, "FoundationPose: Unified 6D pose estimation and tracking of novel objects," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2024, pp. 17 868–17 879.

[36] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *preprint arXiv:1707.06347*, 2017.