Who Moved the Robot? Humanoid Datasets Remember Their Operators

Sihat Afnan, Unnat Jain*, **Habiba Farrukh***
University of California, Irvine
{sihata, unnatj, habibaf}@uci.edu

Abstract

Human-to-humanoid motion retargeting is the central operation in teleoperation data collection: it projects operator demonstrations onto a shared robot skeleton, discarding body-shape parameters, to produce training data for whole-body controllers and foundation models. We report a surprising property of this transform: **operator movement dynamics survive retargeting**. On BONES-SEED (522 operators, 142K sequences), retargeted Unitree G1 trajectories support gender classification at 96.0% and operator re-identification at 97.2% Top-1; on *operators never seen during training*, gender holds at 83.4% and age, height, and weight regress with median absolute error of 2.1 yr, 3.9 cm, and 5.3 kg. Partial correlation analysis reveals emergent, biomechanically interpretable structure: the signals are task-invariant across different activities and persist across different retargeting implementations. We introduce UNVEIL, a spatiotemporal graph convolutional network to measure and interpret this effect. We take initial steps toward understanding how anonymization might help and ask what this implies for data practices and the operator-invariance of learned policies. Project website: https://unveil-humanoid-operators.github.io/.

1 Introduction

Humanoid robot learning is scaling. Our community has built whole-body teleoperation systems that transfer operator motion to humanoids in real time [3, 9, 12, 14, 42], shown that training motion-tracking controllers on motion capture data improves consistently with data volume and compute [2, 21, 42], and is now assembling large shared datasets as training infrastructure for humanoid policies and foundation models.

Motion retargeting is the operation at the center of humanoid data collection. Because a human body and a humanoid like the Unitree G1 differ in morphology and joint count, an operator's motion cannot be executed on the robot directly. Retargeting solves for the robot's joint configuration at each frame by minimizing positional error at a set of shared landmarks (head, hands, feet, torso), subject to joint limits [2, 15, 42]. This discards body-shape parameters entirely: limb proportions, mass distribution, and limb lengths are stripped away, and every operator's motion is projected onto the same robot joint space regardless of their physique (Fig. 1). Crucially, retargeting targets joint configuration, not the dynamics of how joints move over time. Operator movement dynamics shaped by the operator's physiology and motor habits survive retargeting largely intact. The resulting G1 trajectory describes the robot's motion, not the person who produced it; retargeting was designed as a one-way transform. We show it is not entirely one-way. The transform *remembers* its operators: **retargeting removes body morphology yet preserves operator movement dynamics**, and those dynamics are recoverable from the retargeted trajectories alone. Our project website shows a human operator's motion and the corresponding retargeted G1 trajectory.

*Equal Advising.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

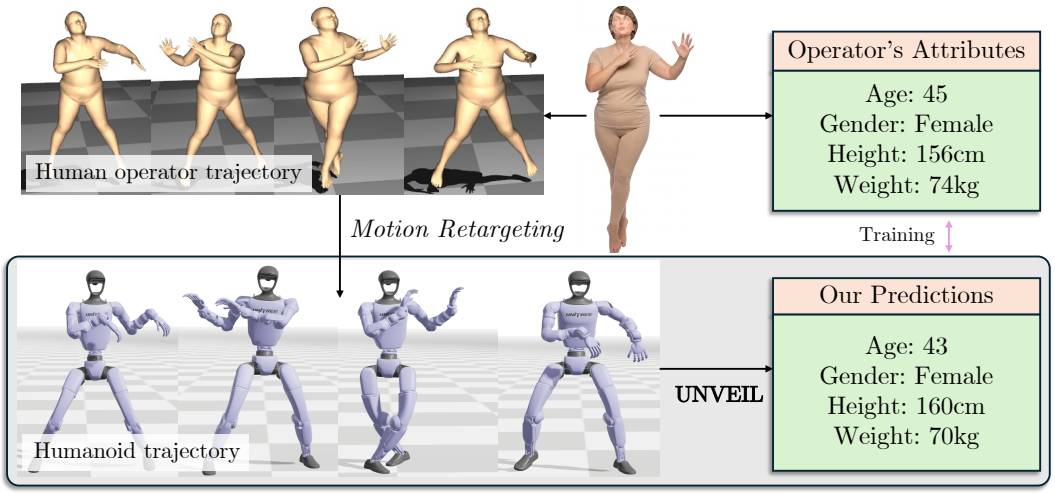


Figure 1: **Humanoid datasets remember their operators.** Retargeting maps each operator onto the same robot skeleton (Unitree G1), discarding body-shape parameters. Retargeting targets joint configuration, not the dynamics of how joints move over time: features of operator movement dynamics persist in the retargeted humanoid trajectories despite body morphology being removed. UNVEIL recovers operator attributes (gender, age, height, weight, and identity) directly from those trajectories, without access to the original human motion.

To measure what survives retargeting, we introduce UNVEIL, a spatiotemporal GCN that encodes position, velocity, and acceleration per joint, models inter-joint dependencies through two hierarchies (intra-limb, limb-torso), and predicts operator attributes from retargeted G1 trajectories. In this sense, UNVEIL operationalizes a tractable and revealing form of *inversion*: recovering operator identity and physique attributes directly from retargeted trajectories, without access to the original human motion. We validate this on BONES-SEED [4]: 142,220 sequences from 522 operators across locomotion, manipulation, dance, and everyday activities, retargeted onto the Unitree G1 with per-operator demographic metadata. On held-out demonstrations from operators seen during training, UNVEIL achieves 96.0% gender classification and 97.2% Top-1 operator re-identification ($\sim$400$\times$ above chance). On operators the model has *never encountered before*, gender classification holds at 83.4% (+33 pp above chance) and age, height, and weight regress with a median absolute error of 2.1 yr, 3.9 cm, and 5.3 kg respectively.

The effect has emergent, biomechanically interpretable structure: joint velocity profiles, ranges of motion, and limb coordination patterns encoding the operator's build survive retargeting and leave consistent traces in the retargeted trajectories. Partial correlation analysis (which isolates each attribute's contribution after accounting for the others) confirms these are genuine independent signals: older operators produce less trunk rotation across every activity category; taller operators consistently cover less root distance. This establishes the leakage as a structural property of the retargeting transformation rather than a dataset artifact. Our project website shows examples of these interpretable features and the statistical evidence behind them.

As teleoperation datasets scale and retargeted trajectories become shared training infrastructure, understanding what they retain about the operators matters both scientifically and practically, for data sharing practices and for the operator-invariance of learned policies.

Our contributions are: **(1)** we identify and characterize operator movement signature leakage in retargeted humanoid trajectories, a previously unexamined property of human-to-humanoid teleoperation, across six inference tasks, 522 operators, and three retargeting implementations (Sec. 4); **(2)** we introduce UNVEIL, a skeleton-aware spatiotemporal GCN that substantially exceeds random and architectural baselines on all tasks, with ablation and partial-correlation analysis grounding the leakage in interpretable, task-invariant motion features (Sec. 3); **(3)** we take steps toward operator anonymization and outline open problems in privacy-aware dataset curation and operator-invariant policy learning, establishing a structure and baseline for future work in our community (Sec. 4).

2 Related Work

Humanoid teleoperation and retargeting. Teleoperation systems that transfer full-body operator motion to humanoids in real time [2, 3, 8, 9, 12, 40, 41] rely on retargeting pipelines that project human motion onto the robot's joint space [2, 14, 15, 42]. SONIC [21] shows that controllers trained on such retargeted data scale favorably with data volume and compute, motivating large shared corpora of retargeted trajectories as humanoid training resources. BONES-SEED [4], the motion-capture corpus central to SONIC's training pipeline, pairs retargeted G1 trajectories with per-operator demographic metadata, making operator attribute inference from retargeted data tractable at scale. That retargeting discards body-shape parameters is well understood; what it preserves about operator movement has not been studied. We study that question.

Skeletal motion understanding. Spatiotemporal GCNs are the standard architecture for motion understanding from joint sequences, from ST-GCN [37] and adaptive two-stream variants [31] to more recent work with learned multi-scale graph structures [7, 10, 36, 17, 20]. The same family has been applied to person re-identification from skeleton data [23, 33], shifting the objective from action classification to performer identification. Unlike prior work in this family, UNVEIL operates on robot-space trajectories rather than human motion, with its graph defined over the robot's kinematic groups and its task being operator attribute recovery rather than action classification. The robot's kinematic tree differs from a human body skeleton in topology and the semantics of each degree of freedom, motivating a graph structure built from the robot's morphology. The learned joint features and limb-torso or intra-limb hierarchical spatial graph design are specific to the retargeting setting: body shape is stripped but movement dynamics survive, and operator-specific information lives in these dynamics, not in body configuration.

Operator movement signatures in human motion. Human motion is biometrically rich: skeleton sequences leak re-identifiable identity and demographic attributes in action-recognition settings [5, 23, 33], and in XR environments users can be re-identified from head and hand motion alone [24, 25, 28, 35]. Retargeting is not merely a change of representation: it clips kinematics to the robot's feasible joint range and maps every operator's motion to the same joint space regardless of physique, making the persistence of operator signature a non-trivial finding. Orthogonal to our focus, prior work has responded with anonymization methods operating on human-space motion directly: adversarial perturbation of joint sequences [23] and retargeting onto dummy skeletons [6]. Whether these signatures survive retargeting onto a robot morphology has not been studied. We show that operator movement signatures do survive retargeting, and that the leakage is structural: it holds across activity types, persists for operators never seen during training, and reflects interpretable patterns in joint velocity and coordination timing.

Toward operator-invariant robot learning. Complementing scaling efforts that have assembled large retargeted datasets for humanoid learning [13, 21, 26], our work raises a question those efforts leave open: whether demonstrator characteristics propagate into trained policies, with implications for generalization and responsible data sharing [29]. If policies trained on retargeted demonstrations learn to rely on operator-specific motion cues, their behavior may not transfer cleanly to new operators or to deployment without a fixed human partner. We characterize this leakage and provide a baseline for operator-invariant policy learning and data curation.

3 UNVEIL

We propose UNVEIL, a spatiotemporal GCN that recovers operator attributes (age, gender, height, weight, identity) from a retargeted humanoid trajectory (Fig. 2). Operator signatures reside in joint coordination, so UNVEIL lifts raw joint states into motion-rich per-joint embeddings (Sec. 3.1) and models coordination at two spatial scales: fine-grained intra-limb coupling and coarser limb-torso coupling (Sec. 3.2). Signatures also span multiple temporal scales, from short bursts of joint activity to slow trajectory-wide trends, so a multi-scale temporal convolution aggregates the per-frame spatial features into a trajectory embedding $\mathbf{z}$ (Sec. 3.3). A prototype reconstruction network distills the learned coordination into an operator-consistent signature, supervised by a contrastive loss; task-specific heads then decode $\mathbf{z}$ into attributes (Sec. 3.4).

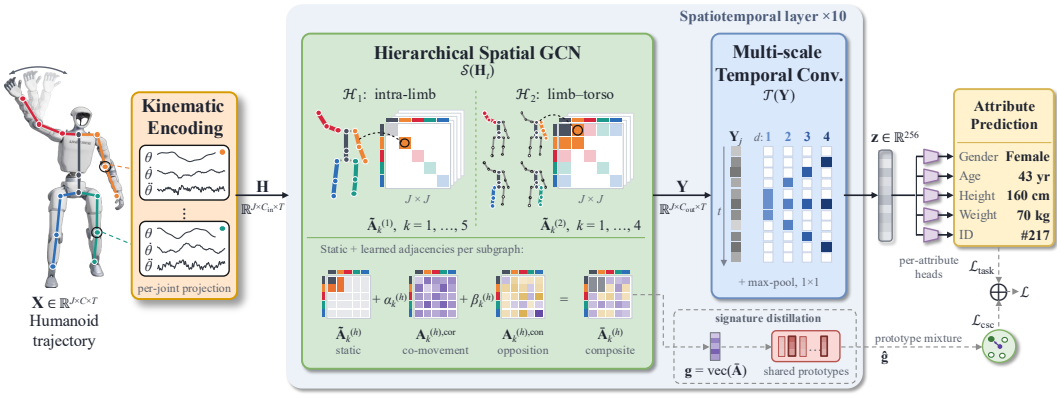


Figure 2: **UNVEIL overview.** Kinematic encoding maps a humanoid trajectory $\mathbf{X}$ to per-joint motion features $\mathbf{H}$, which pass through a hierarchical spatial GCN that models coordination within each limb ($\mathcal{H}_1$) and between each limb and the torso ($\mathcal{H}_2$). Each subgraph combines a static adjacency with learned co-movement and opposition adjacencies, and a multi-scale temporal convolution aggregates the per-frame features $\mathbf{Y}$ into a trajectory embedding $\mathbf{z}$, which per-attribute heads decode under $\mathcal{L}_{\text{task}}$. Signature distillation rebuilds the learned adjacency from shared prototypes under $\mathcal{L}_{\text{csc}}$.

3.1 Kinematic Encoding

We model the humanoid as a kinematic graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of $J = |\mathcal{V}|$ joints and $\mathcal{E}$ connects every joint pair mechanically linked by a rigid body segment. A retargeted trajectory is a sequence of joint configurations $\mathbf{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_T)$, $\mathbf{X}_t \in \mathbb{R}^{J \times C}$, where C is the number of DoF per joint (e.g., $C{=}1$ for the elbow, $C{=}3$ for the shoulder). We partition $\mathcal{V}$ into five groups: torso $\mathcal{V}_{\text{T}}$, left arm $\mathcal{V}_{\text{LA}}$, right arm $\mathcal{V}_{\text{RA}}$, left leg $\mathcal{V}_{\text{LL}}$, and right leg $\mathcal{V}_{\text{RL}}$, corresponding to the kinematic subgraphs in Sec. 3.2. Operator attributes are expressed in how joints move, not where they are: two operators reaching the same target arrive at nearly identical terminal configurations, yet differ in the velocity and acceleration profiles along the way. We map $\mathbf{X} \in \mathbb{R}^{J \times C \times T}$ to $\mathbf{H} \in \mathbb{R}^{J \times C_{\text{in}} \times T}$ with a learned per-joint projection [19, 11], expanding each joint's C raw DoF values into a C_{in}-dimensional feature; the spatial stage then models joint coordination from these embeddings, and the temporal stage captures how coordinated features evolve across frames.

3.2 Hierarchical Spatial Graph Convolution

$\mathbf{H}$ describes each joint independently and does not yet model interactions between joints. Prior spatiotemporal GCNs for skeleton-based motion understanding use a single full-body adjacency at one scale [37, 36, 43, 20]; even data-adaptive variants [7] keep one graph spanning the whole skeleton. Operator signatures, however, are distributed across scales. Adjusting a grip engages only elbow and wrist dynamics within a single arm, whereas reaching across a table also requires the torso to rotate and lean; a single full-body adjacency cannot distinguish these scales. We therefore fix the graph to the robot morphology and decompose $\mathcal{G}$ into a multi-relational set of kinematic subgraphs, each restricting feature propagation to one scale under its own learnable projection (Fig. 2, left).

Intra-limb Hierarchy. Fine-grained motions (e.g., waving) involve coordination between joints within a single limb. Five intra-limb subgraphs $\mathcal{H}_1 = \{\mathcal{G}_k^{(1)}\}_{k=1}^{5}$ (one per body group: torso and four limbs) each contain only the joints and edges within body group k, so graph convolution propagates within the group and remains isolated from all others.

Limb-torso Hierarchy. Larger motions, such as reaching forward to grasp an object, require the limb and the torso to move together, since the torso must rotate or lean to extend the limb's effective range. Four limb-torso subgraphs $\mathcal{H}_2 = \{\mathcal{G}_k^{(2)}\}_{k=1}^{4}$, $k \in \{\text{LA}, \text{RA}, \text{LL}, \text{RL}\}$, each joins limb k's joints with the torso (the shared hub) via the proximal connecting edge; graph convolution propagates between limb k and the torso while remaining isolated from the other limbs.

Adjacency Construction. As shown in Fig. 2 (center), each subgraph $\mathcal{G}_k^{(h)}$ is ultimately represented by three $J\times J$ adjacency matrices: a static and two learned. For each subgraph, we form a binary $J\times J$ adjacency $\mathbf{A}_k^{(h)}$: entries are 1 for connected joint pairs and self-loops within the subgraph, 0 elsewhere (non-member joints have all-zero rows and are not updated). Symmetric normalization $\tilde{\mathbf{A}}_k^{(h)} = D^{-1/2}\mathbf{A}_k^{(h)}D^{-1/2}$ [16] prevents high-degree joints from dominating aggregation; $\tilde{\mathbf{A}}_k^{(h)}$ encodes only the subgraph's fixed skeletal connectivity. Operator signatures, however, are encoded in how joints move relative to one another: during a kick the hand and the knee swing forward despite no physical link, while writing engages one hand and leaves the other still. We therefore augment $\tilde{\mathbf{A}}_k^{(h)}$ with two data-dependent terms [19]: a *correlation adjacency* $\mathbf{A}_k^{(h),\mathrm{cor}}$ for positive joint coupling, and a *contrast adjacency* $\mathbf{A}_k^{(h),\mathrm{con}}$ for negative coupling. Both are computed by projecting per-frame features into query and key streams and then time-averaging over the full trajectory before forming joint-pair scores (details in App. A.2); as a result $\mathbf{A}_k^{(h),\mathrm{cor}}$ and $\mathbf{A}_k^{(h),\mathrm{con}}$ are trajectory-specific: the same learned adjacency is shared across all frames in Eq. (1), capturing the operator's coordination style over the full motion rather than a single instant. They are combined with the static adjacency under learnable scales $\alpha_k^{(h)}$ and $\beta_k^{(h)}$:

$$\text{Composite Adjacency:}\quad \bar{\mathbf{A}}_k^{(h)} = \underbrace{\tilde{\mathbf{A}}_k^{(h)}}_{\text{static skeleton}} + \underbrace{\alpha_k^{(h)}\,\mathbf{A}_k^{(h),\mathrm{cor}}}_{\text{co-movement}} + \underbrace{\beta_k^{(h)}\,\mathbf{A}_k^{(h),\mathrm{con}}}_{\text{opposition}}. \tag{1}$$

Graph Convolution Layer. Each of the nine subgraphs carries its own projection $\mathbf{W}_k^{(h)} \in \mathbb{R}^{C_{\mathrm{in}}\times C_{\mathrm{out}}}$, so the same joint is transformed differently at each scale. The spatial graph convolution on per-frame features $\mathbf{H}_t \in \mathbb{R}^{J\times C_{\mathrm{in}}}$ is:

$$\text{Spatial GCN:}\quad \mathcal{S}(\mathbf{H}_t) = \sigma\left(\sum_{k=1}^{5}\bar{\mathbf{A}}_k^{(1)}\mathbf{H}_t\mathbf{W}_k^{(1)} + \sum_{k=1}^{4}\bar{\mathbf{A}}_k^{(2)}\mathbf{H}_t\mathbf{W}_k^{(2)}\right), \tag{2}$$

where $h{=}1$ sums over the intra-limb subgraphs $\mathcal{H}_1$ and $h{=}2$ over the limb-torso subgraphs $\mathcal{H}_2$.

Operator Signature Distillation. $\bar{\mathbf{A}}_k^{(h)}$ encodes both the operator's consistent coordination style and action-specific motion that varies with each task. We distill the operator-consistent component using a *signature bank* of n_p learnable prototypes, shared templates of joint coordination that recur across trajectories, learned jointly with the network [19, 18, 32, 38, 39] (Fig. 2, bottom). Each entry of $\bar{\mathbf{A}}_k^{(h)}$ describes the coordination between one pair of joints; stacking these joint-pair descriptors into a vector $\mathbf{g}$, the bank reconstructs it as:

$$\text{Signature Distillation:}\quad \hat{\mathbf{g}} = \mathbf{W}_m\,\mathrm{softmax}(\mathbf{W}_q\,\mathbf{g}), \tag{3}$$

where $\mathbf{W}_q$ queries the bank and the softmax weight reflects how well each prototype matches $\mathbf{g}$. Because $\hat{\mathbf{g}}$ is a convex combination of prototypes, action-specific patterns not represented in the bank are discarded; operator-consistent coordination receives high attention weights and is retained. The distilled $\hat{\mathbf{g}}$ drives the operator-contrastive loss $\mathcal{L}_{\mathrm{csc}}$ in Sec. 3.4 (details in App. A.2).

3.3 Temporal Aggregation

The aggregation in Eq. (2) is applied per frame t, combining joint features spatially using the trajectory-level adjacency $\bar{\mathbf{A}}_k^{(h)}$; it produces one spatially-enriched output per frame and does not explicitly model how those outputs evolve across frames. Applying $\mathcal{S}$ to every frame with shared weights yields a per-frame feature tensor $\mathbf{Y} \in \mathbb{R}^{J\times C_{\mathrm{out}}\times T}$, which we then aggregate along the time axis. A single temporal convolution captures dynamics at one temporal scale, but operator signatures manifest at multiple scales: fine-grained joint dynamics over a few frames as well as slow, trajectory-wide tendencies such as peak limb velocity or the rhythm of torso rotation. As illustrated in Fig. 2, we run R temporal convolutions in parallel, each with a distinct dilation rate [20, 7]:

$$\text{Multi-scale Temporal Conv:}\quad \mathcal{T}(\mathbf{Y})_j = \psi\big(\mathrm{concat}_{r=1}^{R}\big(\mathbf{W}_r^{\mathrm{tmp}} *_{d_r} \mathbf{Y}_j\big)\big), \tag{4}$$

where $\mathbf{Y}_j \in \mathbb{R}^{C_{\mathrm{out}}\times T}$ is joint j's time series; $*_{d_r}$ is 1-D convolution with dilation d_r (the kernel skips $d_r{-}1$ frames between taps, widening the temporal receptive field without adding parameters); concat stacks the R branch outputs along the channel axis; and ψ is a learned linear projection. Setting $d_r{=}1$ on one branch captures short bursts of joint activity; larger d_r on other branches captures slow trends spanning much of the trajectory, all within one layer at the same parameter budget.

3.4 Architecture and Training

UNVEIL stacks spatiotemporal layers, each composing the spatial graph convolution (Sec. 3.2) and the multi-scale temporal convolution (Sec. 3.3) with batch normalization, a residual connection, and dropout. Early layers operate at full temporal resolution and capture fine-grained per-frame features; two strided intermediate layers progressively widen the temporal receptive field. A global average pool over the joints and time dimensions yields trajectory embedding $\mathbf{z}$, which task-specific heads decode into operator attributes. Architecture and training hyperparameters are in Tab. A.2 (App. A.2).

Training. We train a separate UNVEIL instance per attribute. The trajectory embedding $\mathbf{z}$ is decoded by a task-specific head: a softmax classifier for gender and identity, and a linear regressor for age, height, and weight. For each attribute we minimize a task loss augmented with the operator-contrastive term on the distilled $\hat{\mathbf{g}}$ from Sec. 3.2 [19]:

$$\text{Training Objective:} \quad \mathcal{L} \;=\; \mathcal{L}_{\text{task}} \;+\; \lambda_{\text{csc}}\, \mathcal{L}_{\text{csc}} \tag{5}$$

where $\mathcal{L}_{\text{task}}$ is cross-entropy with label smoothing for classification (gender, identity) and MSE for regression (age, height, weight). The contrastive term $\mathcal{L}_{\text{csc}}$ is activated after a warm-up period; all other hyperparameters are listed in Tab. A.2 and detailed in App. A.2.

4 Experiments

In this section, we describe UNVEIL's evaluation setup (Sec. 4.1), present quantitative results across six inference tasks and three retargeting implementations (Sec. 4.2), report ablations and interpretability analyses that characterize the emergence of operator motion patterns in the retargeted trajectories (Sec. 4.3 and Sec. 4.4), and ask what the leakage means for the operators who provide the data and whether it can be reduced (Sec. 4.5).

4.1 Setup

BONES-SEED Dataset [4]. A large-scale motion-capture dataset from [21] containing 142,220 G1-retargeted trajectories from 522 operators across 20 activity categories (locomotion, dance, sports, everyday activities, etc.). Each *demonstration* is one operator performing one activity, available as both a human motion trajectory and a G1 joint-angle trajectory. Each operator carries a pseudonymous label, which we treat as that operator's identity, and the demographic metadata released with the dataset.[2] Operators span ages 17–71 yr, heights 145–199 cm, weights 38–145 kg, with a balanced gender split (distributions and G1 joint layout in App. A.1). This pairing of retargeted trajectories with operator labels, across many operators and activities, enables us to study whether retargeted humanoid datasets remember their operators.

Two axes of generalization. A retargeted dataset presents two distinct kinds of novelty at inference: a known operator captured in a new demonstration, or an operator absent from training entirely. We evaluate both by constructing two train/test splits from the same pool of operators.

- The *unseen-demos-of-seen-operators* protocol holds out a disjoint set of *demonstrations* from operators who also appear in training. Every test operator is known, but each held-out demonstration is unseen, so it isolates generalization across demonstrations: can UNVEIL recognize a known operator's signature in a demonstration it was not trained on?
- The *unseen-operators* protocol holds out a disjoint set of *operators*, whose demonstrations never appear in training. Both the operators and their demonstrations are new, so it isolates generalization across the population: do the learned attribute cues transfer to people UNVEIL has never encountered?

Re-identification and linkage are defined only under the unseen-demos-of-seen-operators protocol, since matching a query to a known operator requires that operator to be enrolled during training; attribute inference (gender, age, height, weight) is evaluated under both.

Tasks and metrics. We evaluate UNVEIL on six inference tasks that recover operator attributes from retargeted trajectories. Gender classification reports accuracy. Age, height, and weight are continuous regression tasks evaluated by median absolute error (MedAE). Re-identification matches a query trajectory against a per-operator gallery of held-out demos via cosine similarity in the learned

[2] We make no attempt to link any trajectory, label, or attribute to a real person, in keeping with the dataset's terms of use.

Table 1: UNVEIL's performance across inference tasks, evaluation settings, and retargeting implementations. SOMA is the retargeting pipeline used by BONES-SEED; OmniRetarget and GMR are applied to the same human motion data to produce independently retargeted G1 trajectories. Leakage persists across all three.

	Task → / Evaluation Setting ↓	Gender Acc(%)↑	Weight MedAE↓	Height MedAE↓	Age MedAE↓	Re-ID Top-1 | Top-5 Acc(%)↑	Linkage Acc(%)↑
SOMA [30]	Unseen operators	83.4	5.3 kg	3.9 cm	2.1 yr	—	—
	Unseen demos of seen operators	96.0	1.9 kg	1.9 cm	1.6 yr	97.2 | 98.8	92.4
OmniRetarget [40]	Unseen operators	80.9	6.3 kg	5.1 cm	1.8 yr	—	—
	Unseen demos of seen operators	94.6	4.3 kg	3.2 cm	1.7 yr	90.5 | 95.6	88.7
GMR [2]	Unseen operators	76.1	8.4 kg	5.7 cm	2.6 yr	—	—
	Unseen demos of seen operators	91.0	4.2 kg	4.1 cm	1.9 yr	90.8 | 97.2	90.2

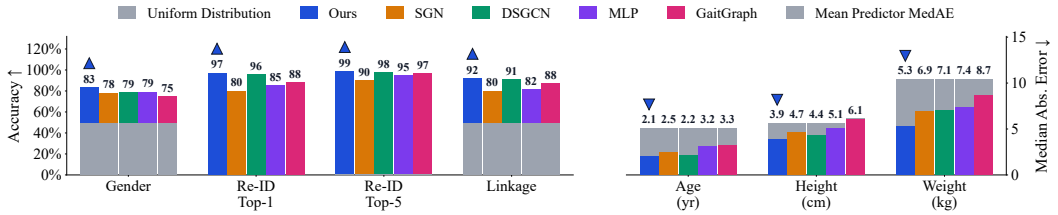


Figure 3: **UNVEIL outperforms four baselines spanning three modeling families**: adapted spatial-graph designs (SGN, DS-GCN), a simple joint-feature MLP that treats frames independently, and GaitGraph, a skeleton-based gait-recognition model. Left: classification accuracy (gender, re-identification Top-1/Top-5, linkage); right: regression error as median absolute error (medAE, ↓) for age, height, and weight. UNVEIL consistently leads on every task

embedding space, reporting Top-1 and Top-5 accuracy. Linkage decides whether two trajectories were produced by the same operator by thresholding their pairwise cosine similarity, reporting accuracy.

Calibrating a first-of-its-kind result. This work provides the first evidence of operator attributes leaking through retargeted humanoid trajectories collected for robot learning. While no direct baselines exist, we calibrate our results in two ways. First, we report intuitive task-specific chance references: 50% accuracy for gender and linkage, the MedAE of the mean predictor for regression tasks, and $1/N_{\text{operators}}$ for re-identification. Second, since no prior method targets operator inference on robot trajectories, we build baselines from three skeleton-based architectures developed for human motion, GaitGraph [33] from gait recognition, and SGN [43] and DSGCN [36] from action recognition, together with a two-layer MLP over motion features as a baseline without skeleton structure. Each replaces UNVEIL's hierarchical spatial convolution and temporal aggregation modules with its own architecture, keeps the task-specific heads (Sec. 3.4), and is trained from scratch on retargeted G1 trajectories under the same splits and protocol. We adapt code from the official releases, with implementation details in App. A.3.

4.2 Quantitative Results

Humanoid datasets *do* carry operator attributes (Tab. 1, Fig. 3). From humanoid trajectories alone, UNVEIL infers gender at 83.4% accuracy on operators never seen during training, 33.4 pp above chance. For continuous attributes under the same unseen-operators protocol, UNVEIL attains a median absolute error (MedAE) of 2.1 yr for age, 3.9 cm for height, and 5.3 kg for weight; that is, for at least half of these previously unseen operators, UNVEIL predicts age, height, and weight to within roughly two years, four centimeters, and five kilograms, respectively. On seen operators with held-out demonstrations, gender accuracy rises to 96.0% and the regression errors decrease further.

Retargeted trajectories function as soft biometrics for operator identity (Tab. 1). UNVEIL re-identifies the operator behind a retargeted G1 trajectory with 97.2% Top-1 accuracy, roughly $400\times$ the $1/N_{\text{operators}}$ chance baseline of 0.2%. The same embedding supports operator linkage, deciding whether two trajectories belong to the same operator, at 92.4% accuracy against a 50%

Table 2: Ablation study of UNVEIL. Each row removes one component (marked ✗) while keeping the other two (marked ✓). ↓ indicates the change from the full model. The first row is our full model.

UNVEIL Components			Gender Accuracy(%)		Re-ID Accuracy(%)	
Kinematic Encoding	Hierarchical Spatial Graph	Temporal Aggregation	Seen Operator Unseen Demo	Unseen Operator	Top-1	Top-5
✓	✓	✓	**96.0**	**83.4**	**97.2**	**98.8**
✗	✓	✓	94.1 (↓1.9)	81.2 (↓2.2)	93.0 (↓4.0)	98.1 (↓0.7)
✓	✗	✓	93.9 (↓2.1)	81.3 (↓2.1)	94.7 (↓2.5)	98.0 (↓0.8)
✓	✓	✗	87.7 (↓8.3)	76.9 (↓6.5)	86.7 (↓10.5)	94.7 (↓4.1)

chance baseline. This near-saturated performance indicates that retargeted humanoid trajectories carry operator signatures strong enough to serve as soft biometrics.

Leakage persists across retargeting implementations (Tab. 1). Tab. 1 reports results for three retargeting pipelines applied to the same human motion data: SOMA (the frame-by-frame IK shipped with BONES-SEED), OmniRetarget [40] (which enforces interaction and contact constraints), and GMR [2] (which uses two-stage inverse kinematics at each frame). Despite these different retargeting formulations, operator attributes remain recoverable: unseen-operator gender prediction stays at 80.9% and 76.1%, and Top-1 re-identification at 90.5% and 90.8% for OmniRetarget and GMR, respectively. This shows that the leakage is a structural property of the retargeting transform, not an artifact of any single implementation.

Findings generalize across modeling choices (Fig. 3, App. A.3). To verify that the leakage is intrinsic to the retargeted trajectories rather than dependent on UNVEIL's architecture, we evaluate four baselines with different architectures (Sec. 4.1). A two-layer MLP over hand-crafted motion features, with no graph structure or temporal modeling, already recovers 85.3% Top-1 re-identification and 79.4% unseen-operator gender. GaitGraph [33], transferred from a different domain, reaches 88.4% Top-1 re-identification. SGN [43] and DSGCN [36] perform comparably. All four recover operator attributes well above chance across every task (Fig. 3), showing that the operator movement dynamics can be recovered independent of the model. UNVEIL's margin comes from jointly modeling spatial joint dependencies and temporal dynamics, which matters most in the harder unseen-operator setting where the gap widens (83.4% vs. 74.8% gender for GaitGraph).

4.3 Ablations

We ablate each of UNVEIL's three components by removing it while keeping the other two, retraining from scratch on the same splits (Tab. 2).

- *Motion-derivative features outperform raw joint angles.* Bypassing the kinematic encoder and feeding raw joint angles instead of the encoded velocity and acceleration features costs $\sim$2 pp gender and 4 pp Top-1 re-identification, locating operator signal in per-joint motion derivatives rather than static configuration.
- *Anatomical subgraphs outperform a single adjacency.* Replacing the intra-limb and limb-torso subgraphs with one fully-connected graph over all joints drops gender accuracy by $\sim$2 pp and Top-1 re-identification by 2.5 pp, showing separate per-subgraph weights resolve coordination a shared adjacency cannot.
- *Temporal aggregation is the most critical component.* Collapsing each demonstration to its mean frame yields the largest drops, gender by 8.3 pp and Top-1 re-identification by 10.5 pp, confirming UNVEIL's premise that operator signatures reside in temporal dynamics rather than any single frame.

4.4 What Survives Retargeting

Retargeting discards body shape but not the temporal profile of motion. It solves for joint configurations that reproduce the operator's pose frame by frame and places no constraint on how those configurations evolve over time, so velocity profiles, ranges of motion, and coordination timing, which depend on the operator's physiology, pass into the robot trajectory. To examine what this preserves, we extract 27 interpretable features per demonstration: the mean velocity, peak velocity, and range of motion of nine body regions (details in App. A.4).

Table 3: Operator attributes leave distinct traces in robot motion. For each attribute, the three motion features with the strongest correlation, averaged over activities. r is the Pearson correlation between the feature and the attribute across operators performing the same activity; r_{partial} controls for the other three attributes.

Attribute	Feature	r	r_{partial}
Age	`waist_rom`	-0.30	-0.25
	`hip_peak_vel`	-0.29	-0.28
	`waist_mean_vel`	-0.27	-0.23
Height	`root_translate_rom`	-0.25	-0.25
	`root_translate_peak_vel`	-0.19	-0.18
	`ankle_peak_vel`	-0.16	-0.14
Weight	`root_translate_peak_vel`	-0.25	-0.20
	`waist_rom`	-0.20	-0.14
	`wrist_mean_vel`	-0.15	-0.08
Gender	`ankle_rom`	-0.31	-0.18
	`wrist_rom`	-0.14	-0.13
	`shoulder_rom`	-0.11	-0.10

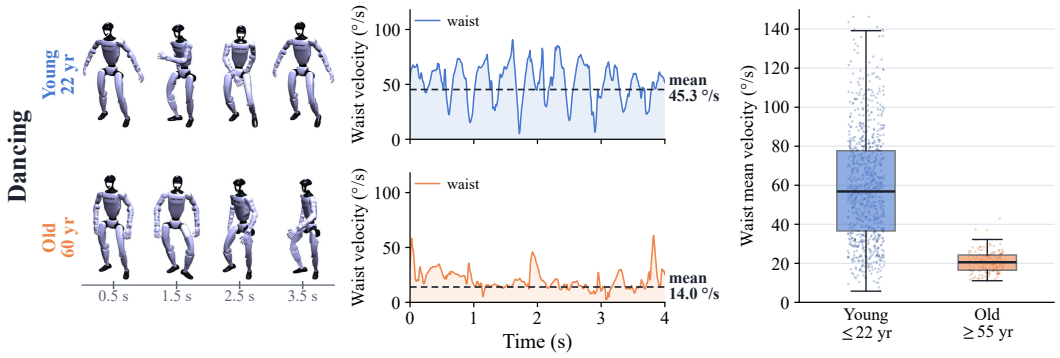


Figure 4: **Operator age survives retargeting.** *Left and center:* operators aged 22 and 60 yr both perform dancing; the robot operated by the older operator rotates its waist more slowly. *Right:* waist mean velocity over all dancing demonstrations for young ($\leq$22 yr) and old ($\geq$55 yr) operators; each dot is one demonstration.

Operator attributes correlate with interpretable motion features (Tab. 3). We compute the Pearson correlation r between each feature and each attribute across operators performing the same activity. Tab. 3 lists the three most strongly correlated features per attribute. A negative correlation means that the feature decreases as the attribute increases. For gender, which we code as 0 for female and 1 for male, a negative value means that the feature is larger for female operators. Since the four attributes are mutually correlated in the population (e.g., weight may correlate with height), we also report the partial correlation r_{partial}, which measures how strongly a feature relates to one attribute after accounting for the other three. If r_{partial} stays close to r, the feature tracks that attribute itself. If it shrinks toward zero, the raw correlation was mostly inherited from the other attributes. Several correlations persist:

- *Older operators move their torso less.* Waist range of motion remains correlated with age ($r_{\text{partial}}{=}-0.25$), consistent with reduced spinal mobility.
- *Taller operators cover less root distance per unit time.* Root translation range remains correlated with height (-0.25, equal to its raw value).
- *Heavier operators produce slower root motion.* Root peak velocity remains correlated with weight but decreases from -0.25 to -0.20.

Feature correlation generalizes across population (Fig. 4). Fig. 4 (left) compares two operators of different ages performing the same activity. While dancing, the robot operated by the older operator rotates its waist more slowly (mean 14.0 vs. 45.3 °/s). The box plots (Fig. 4, right) show that across all demonstrations of the same activity, operators aged 55 and older have consistently lower waist

Table 4: Anonymization reduces leakage while retaining motion utility. Attributes are inferred by UNVEIL; tracking uses a frozen SONIC policy; BC-RNN reports mean tail success.

	Operator Attributes						Utility		
	Re-ID Top-1↓	Linkage Acc↓	Gender Acc↓	Age MedAE↑	Height MedAE↑	Weight MedAE↑	Action Top-1↑	Tracking Success↑	BC-RNN Success↑
Original	97.2%	92.4%	83.4%	2.1 yr	3.9 cm	5.3 kg	88.7%	91.8%	33.6%
Anonymized	16.8%	53.1%	51.8%	5.4 yr	7.7 cm	8.5 kg	80.0%	72.3%	35.9%

mean velocity than operators aged 22 and younger. The age signature is therefore a property of the retargeted data at the population level. Our project website visualizes these features on individual demonstrations and for other attributes.

4.5 Implications for Humanoid Learning

Prospective operators care about the leakage. We conducted a 100-participant survey on Prolific [1] to measure whether the leakage changes prospective operators' willingness to contribute teleoperation data.[3] Each participant was asked to imagine joining a crowdsourced robot learning program in which their retargeted robot trajectories are shared publicly, and to report their willingness to contribute before and after a disclosure that such trajectories can reveal operator attributes. Before the disclosure, 73% believed that retargeting conceals the operator. After it, 63% of initially willing participants withdrew. Among participants unwilling after the disclosure, a technical guarantee that operator attributes cannot be recovered from the released trajectories was the most preferred mitigation (75%).

An initial anonymizer reduces attribute recovery at a measurable utility cost (Tab. 4). As an initial step toward mitigating leakage, we explore whether we can remove the operator signature from a trajectory while preserving its motion content. We disentangle each trajectory into a motion factor and an operator factor with two encoders and a shared decoder, trained adversarially against UNVEIL, and anonymize by decoding the motion factor with the operator factor of a demographically distant operator (details in App. A.5). Anonymization substantially reduces attribute recovery. Re-ID Top-1 decreases from 97.2% to 16.8%, linkage and gender accuracy fall to within a few points of their 50% chance floors, and age, height, and weight errors roughly double. However, it also reduces utility. Activity recognition Top-1 decreases by 8.7 pp and tracking success from 91.8% to 72.3%. The reduction in tracking success concentrates in contact-rich motion. For instance, on activities involving object manipulation, tracking success decreases from 96.4% to 55.4% (App. A.5). Anonymization thus offers a path to releasing crowdsourced or retargeted data publicly, but at the cost of motion fidelity. Until that cost is reduced, access control and data governance remain necessary.

5 Conclusion

We have taken a first step toward characterizing and addressing operator privacy in retargeted humanoid datasets, and provide the structure necessary for advancing future work along several axes. Our current analysis focuses on a single robot morphology, so extending it to platforms with different joint layouts is the natural next step, as is asking whether operator signatures propagate into the policies trained on retargeted data. Where they do, the answer is not always to remove them. In personalization and natural human-robot interaction, operator style is the object of study, and consent and access control are the instruments for protecting the operators who provide it. Where data is released publicly, anonymization becomes necessary, and our baseline, trained without a physics objective, leaves room for generative alternatives [27, 34] that preserve motion fidelity better. Understanding how working operators and data collectors actually handle operator privacy is what will turn our findings into practice. We have released our source code and study protocols to support these directions.

[3] The study was reviewed by our IRB and determined to be exempt.

Acknowledgments

We thank the NeurIPS reviewers and AC for their valuable feedback. This work uses Motion Data by Bones Studio (https://bones.studio/). We thank Bones Studio for creating and releasing the BONES-SEED dataset [4], whose scale and per-operator metadata made this study possible. This work was partially supported by the National Science Foundation (NSF) under grant CNS-2544561 and by the Department of Computer Science in the Donald Bren School of Information and Computer Sciences at the University of California, Irvine. The findings and conclusions presented in this paper are solely those of the authors and do not necessarily represent the views of the NSF.

References

[1] Prolific. https://www.prolific.com. Online participant recruitment platform. Accessed November 2026.

[2] Joao Pedro Araujo, Yanjie Ze, Pei Xu, Jiajun Wu, and C. Karen Liu. Retargeting matters: General motion retargeting for humanoid motion tracking. In *IEEE International Conference on Robotics and Automation*, 2026.

[3] Qingwei Ben, Feiyu Jia, Jia Zeng, Junting Dong, Dahua Lin, and Jiangmiao Pang. HOMIE: Humanoid loco-manipulation with isomorphic exoskeleton cockpit. In *Robotics: Science and Systems*, 2025.

[4] BONES Studio. BONES-SEED: Skeletal everyday embodiment dataset. URL https://huggingface.co/datasets/bones-studio/seed. Accessed: November 06, 2026.

[5] Thomas Carr, Aidong Lu, and Depeng Xu. Linkage attack on skeleton-based motion visualization. In *ACM International Conference on Information and Knowledge Management*, 2023.

[6] Thomas Carr, Depeng Xu, and Aidong Lu. Adversary-guided motion retargeting for skeleton anonymization. *arXiv preprint arXiv:2405.05428*, 2024.

[7] Yuxin Chen, Ziqi Zhang, Chunfeng Yuan, Bing Li, Ying Deng, and Weiming Hu. Channel-wise topology refinement graph convolution for skeleton-based action recognition. In *IEEE/CVF International Conference on Computer Vision*, 2021.

[8] Xuxin Cheng, Yandong Ji, Junming Chen, Ruihan Yang, Ge Yang, and Xiaolong Wang. Expressive whole-body control for humanoid robots. In *Robotics: Science and Systems*, 2024.

[9] Xuxin Cheng, Jialong Li, Shiqi Yang, Ge Yang, and Xiaolong Wang. Open-TeleVision: Teleoperation with immersive active visual feedback. In *Conference on Robot Learning*, 2024.

[10] Hyung-gun Chi, Myoung Hoon Ha, Seunggeun Chi, Sang Wan Lee, Qixing Huang, and Karthik Ramani. Infogcn: Representation learning for human skeleton-based action recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[11] Haodong Duan, Jiaqi Wang, Kai Chen, and Dahua Lin. PySKL: Towards good practices for skeleton action recognition. In *ACM International Conference on Multimedia*, 2022.

[12] Zipeng Fu, Qingqing Zhao, Qi Wu, Gordon Wetzstein, and Chelsea Finn. Humanplus: Humanoid shadowing and imitation from humans. In *Conference on Robot Learning*, 2024.

[13] Zipeng Fu, Tony Z. Zhao, and Chelsea Finn. Mobile ALOHA: Learning bimanual mobile manipulation with low-cost whole-body teleoperation. In *Conference on Robot Learning*, 2024.

[14] Tairan He, Zhengyi Luo, Xialin He, Wenli Xiao, Chong Zhang, Weinan Zhang, Kris Kitani, Changliu Liu, and Guanya Shi. OmniH2O: Universal and dexterous human-to-humanoid whole-body teleoperation and learning. In *Conference on Robot Learning*, 2024.

[15] Tairan He, Zhengyi Luo, Wenli Xiao, Chong Zhang, Kris Kitani, Changliu Liu, and Guanya Shi. Learning human-to-humanoid real-time whole-body teleoperation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2024.

[16] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017.

[17] Jungho Lee, Minhyeok Lee, Dogyoon Lee, and Sangyoun Lee. Hierarchically decomposed graph convolutional networks for skeleton-based action recognition. In *IEEE/CVF International Conference on Computer Vision*, 2023.

[18] Minhyeok Lee, Suhwan Cho, Seunghoon Lee, Chaewon Park, and Sangyoun Lee. Unsupervised video object segmentation via prototype memory network. In *IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023.

[19] Hongda Liu, Yunfan Liu, Min Ren, Hao Wang, Yunlong Wang, and Zhenan Sun. Revealing key details to see differences: A novel prototypical perspective for skeleton-based action recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

[20] Ziyu Liu, Hongwen Zhang, Zhenghao Chen, Zhiyong Wang, and Wanli Ouyang. Disentangling and unifying graph convolutions for skeleton-based action recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[21] Zhengyi Luo, Ye Yuan, Tingwu Wang, Chenran Li, Fernando Castañeda, Sirui Chen, Zi-Ang Cao, Jiefeng Li, David Minor, Qingwei Ben, et al. Sonic: Supersizing motion tracking for natural humanoid whole-body control. *Science Robotics*, 2026.

[22] Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. In *Conference on Robot Learning (CoRL)*, 2021.

[23] Saemi Moon, Myeonghyeon Kim, Zhenyue Qin, Yang Liu, and Dongwoo Kim. Anonymization for skeleton action recognition. In *AAAI Conference on Artificial Intelligence*, 2023.

[24] Vivek Nair, Wenbo Guo, Justus Mattern, Rui Wang, James F. O'Brien, Louis Rosenberg, and Dawn Song. Unique identification of 50,000+ virtual reality users from head and hand motion data. In *USENIX Security Symposium*, 2023.

[25] Vivek C. Nair, Gonzalo Munilla-Garrido, and Dawn Song. Going incognito in the metaverse: Achieving theoretically optimal privacy-usability tradeoffs in vr. In *ACM Symposium on User Interface Software and Technology*, 2023.

[26] Open X-Embodiment Collaboration. Open X-Embodiment: Robotic learning datasets and RT-X models. In *IEEE International Conference on Robotics and Automation*, 2024.

[27] Mathis Petrovich, Michael J. Black, and Gül Varol. Action-conditioned 3D human motion synthesis with transformer VAE. In *IEEE/CVF International Conference on Computer Vision*, 2021.

[28] Ken Pfeuffer, Matthias J. Geiger, Sarah Prange, Lukas Mecke, Daniel Buschek, and Florian Alt. Behavioural biometrics in VR: Identifying people from body motion and relations in virtual reality. In *ACM Conference on Human Factors in Computing Systems*, 2019.

[29] Maria Rigaki and Sebastian Garcia. A survey of privacy attacks in machine learning. *ACM Computing Surveys*, 2023.

[30] Jun Saito, Jiefeng Li, Michael de Ruyter, Miguel Guerrero, Edy Lim, Ehsan Hassani, Roger Blanco Ribera, Hyejin Moon, Magdalena Dadela, Marco Di Lucca, et al. Soma: Unifying parametric human body models. *arXiv preprint arXiv:2603.16858*, 2026.

[31] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[32] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, 2017.

[33] Torben Teepe, Ali Khan, Johannes Gilg, Fabian Herzog, Stefan Hörmann, and Gerhard Rigoll. GaitGraph: Graph convolutional network for skeleton-based gait recognition. In *IEEE International Conference on Image Processing*, 2021.

[34] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H. Bermano. Human motion diffusion model. In *International Conference on Learning Representations*, 2023.

[35] Pier Paolo Tricomi, Federica Nenna, Luca Pajola, Mauro Conti, and Luciano Gamberini. You can't hide behind your headset: User profiling in augmented and virtual reality. *IEEE Access*, 2023.

[36] Jianyang Xie, Yanda Meng, Yitian Zhao, Anh Nguyen, Xiaoyun Yang, and Yalin Zheng. Dynamic semantic-based spatial graph convolution network for skeleton-based human action recognition. In *AAAI Conference on Artificial Intelligence*, 2024.

[37] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *AAAI Conference on Artificial Intelligence*, 2018.

[38] Boyu Yang, Chang Liu, Bohao Li, Jianbin Jiao, and Qixiang Ye. Prototype mixture models for few-shot semantic segmentation. In *European Conference on Computer Vision*, 2020.

[39] Hong-Ming Yang, Xu-Yao Zhang, Fei Yin, and Cheng-Lin Liu. Robust classification with convolutional prototype learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

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

[41] Taerim Yoon, Dongho Kang, Seungmin Kim, Jin Cheng, Min Sung Ahn, Stelian Coros, and Sungjoon Choi. Spatio-temporal motion retargeting for quadruped robots. *IEEE Transactions on Robotics*, 2025.

[42] Yanjie Ze, Zixuan Chen, João Pedro Araújo, Zi ang Cao, Xue Bin Peng, Jiajun Wu, and C. Karen Liu. TWIST: Teleoperated whole-body imitation system. In *Conference on Robot Learning*, 2025.

[43] Pengfei Zhang, Cuiling Lan, Wenjun Zeng, Junliang Xing, Jianru Xue, and Nanning Zheng. Semantics-guided neural networks for efficient skeleton-based human action recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

A Appendix

A.1: Dataset details including BONES-SEED attribute distributions and the Unitree G1 joint layout.

A.2: Training details including hyperparameters, architecture specifics, and compute requirements.

A.3: Additional baselines and architecture comparisons. We adapt four architectures, MLP, GaitGraph [33], SGN [43], and DSGCN [36] to our pipeline under the same evaluation protocol as Tab. 1, showing that the leakage is intrinsic to the data rather than dependent on UNVEIL's design.

A.4: Correlation analysis details covering feature extraction, aggregation, and the raw and partial correlation analysis.

A.5: Anonymization method and objective, SONIC tracking and motion quality, behavioral cloning equity, and transfer of the privacy reduction to architectures the anonymizer was not trained against.

A.6: User study protocol and the survey questions shown to participants.

A.1 Dataset Details

Attribute Distributions. We report the distribution of the four operator attributes in Fig. A.1. The dataset spans ages 17–71 yr, heights 145–199 cm, weights 38–145 kg, and has an approximately balanced gender split (241 F, 251 M).

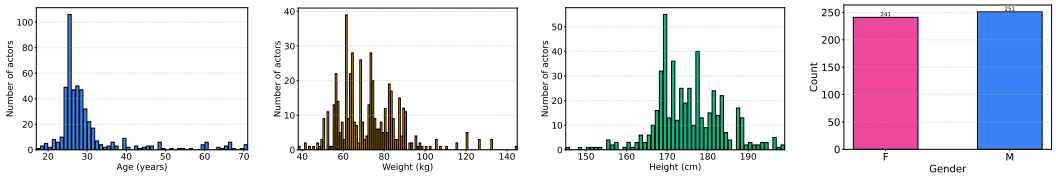


Figure A.1: Distribution of operator attributes in the dataset, including age, height, weight, and gender.

G1 Retargeted Trajectory Format. Each retargeted trajectory is a sequence of $J{=}15$ joint configurations with 35 total channels, sampled at 120 Hz. Tab. A.1 details the kinematic layout. The root provides 6 degrees of freedom (3 translational, 3 rotational); the remaining 29 channels are joint angles distributed across 13 actuated joints grouped into five kinematic chains: left leg (hip, knee, ankle), right leg (hip, knee, ankle), torso (waist, root), left arm (shoulder, elbow, wrist), and right arm (shoulder, elbow, wrist).

A.2 Training Details

We construct the static adjacency from the Unitree G1 kinematic tree and group its 35 control DoFs into $J{=}15$ joints; joints with fewer than three rotational DoFs are zero-padded so that every joint carries 3 input features. The trajectory embedding $\mathbf{z}$ is decoded by a softmax classifier over the 393 training operators for re-identification, a two-way softmax for gender, and a linear regressor for age, height, and weight. All experiments run on a single NVIDIA RTX 4090 GPU (24 GB), and one epoch over the training split takes about 2000 seconds.

Kinematic subgraph structure. The intra-limb hierarchy ($h{=}1$) contains one subgraph $\mathcal{G}_k^{(1)} = (\mathcal{V}_k, \mathcal{E}_k)$ for each body group $k \in \{\mathrm{T}, \mathrm{LA}, \mathrm{RA}, \mathrm{LL}, \mathrm{RL}\}$, restricted to the joints and edges within that group. The limb-torso hierarchy ($h{=}2$) contains one subgraph $\mathcal{G}_k^{(2)} = (\mathcal{V}_k \cup \mathcal{V}_\mathrm{T},\ \mathcal{E}_k \cup \mathcal{E}_\mathrm{T} \cup \{e_k\})$ for each limb $k \in \{\mathrm{LA}, \mathrm{RA}, \mathrm{LL}, \mathrm{RL}\}$, where $e_k \in \mathcal{E}$ is the edge connecting the limb's proximal joint to the torso. Each binary adjacency $\mathbf{A}_k^{(h)}$ is defined over all J joints, with entries 1 for connected pairs and self-loops within the subgraph and 0 elsewhere, so joints outside the subgraph are not updated by it.

Table A.1: Unitree G1 retargeted joint layout. Each row specifies a joint, its degrees of freedom, and the kinematic group it belongs to for UNVEIL's graph construction.

Joint	Channels	DoF	Kinematic Group
Root translation	3	translateX/Y/Z	Torso
Root rotation	3	rotateX/Y/Z	Torso
Left hip	3	pitch, roll, yaw	Left leg
Left knee	1	pitch	Left leg
Left ankle	2	pitch, roll	Left leg
Right hip	3	pitch, roll, yaw	Right leg
Right knee	1	pitch	Right leg
Right ankle	2	pitch, roll	Right leg
Waist	3	pitch, roll, yaw	Torso
Left shoulder	3	pitch, roll, yaw	Left arm
Left elbow	1	pitch	Left arm
Left wrist	3	pitch, roll, yaw	Left arm
Right shoulder	3	pitch, roll, yaw	Right arm
Right elbow	1	pitch	Right arm
Right wrist	3	pitch, roll, yaw	Right arm
Total	**35**		

Adaptive adjacency and signature bank. Within each subgraph, two learned projections map the per-frame features to query and key streams, which are averaged over time to give one embedding per joint, $\mathbf{Q}$ and $\mathbf{K}$. Dropping the subgraph indices, the correlation and contrast adjacencies of Eq. (1) are

$$\mathbf{A}^{\mathrm{cor}} = \mathrm{softmax}\big(\mathbf{Q}\mathbf{K}^{\top}\big)\,, \qquad \mathbf{A}^{\mathrm{con}} = \tanh(\mathbf{Q} \boxminus \mathbf{K})\,, \tag{A.1}$$

Here, $\boxminus$ computes the difference between every query–key pair separately for each feature channel, producing one contrast adjacency matrix per channel. The scales $\alpha_k^{(h)}$ and $\beta_k^{(h)}$ are initialized to zero, so training starts from the static adjacency $\tilde{\mathbf{A}}_k^{(h)}$. The signature bank of Eq. (3) holds n_p=100 learned prototypes, stored as the columns of $\mathbf{W}_m$. We flatten the final layer's composite adjacency into g. The signature bank reconstructs g as a weighted average of its 100 prototypes, using the softmax weights in Eq. (3).

Class-specific contrastive loss. The class-specific contrastive loss $\mathcal{L}_{\mathrm{csc}}$ of Eq. (5) follows the prototype learning network [19]: instead of contrasting samples against one another, it contrasts each sample against a bank of momentum-updated *class prototypes*, tightening the distilled signature $\hat{\mathbf{g}}$ around its class and pushing classes apart. We collapse $\hat{\mathbf{g}}$ to a contrastive feature $\mathbf{f} \in \mathbb{R}^H$ (H=256) by channel-wise average pooling followed by a learnable linear map, and ℓ_2-normalize it. The bank $\{\mathbf{m}_k\}_{k=1}^K$ stores one prototype per class; each is a momentum estimate of its class centroid, updated every step from the batch mean $\bar{\mathbf{f}}_k$ of the class-k features,

$$\mathbf{m}_k \;\leftarrow\; \mu\,\mathbf{m}_k + (1-\mu)\,\bar{\mathbf{f}}_k, \qquad \mu = 0.9. \tag{A.2}$$

$\mathcal{L}_{\mathrm{csc}}$ is the resulting prototypical InfoNCE, treating the sample's own prototype $\mathbf{m}_k$ as the sole positive and the remaining $\mathbf{m}_{l\neq k}$ as negatives,

$$\mathcal{L}_{\mathrm{csc}} = -\,\log \frac{\exp\big(\mathbf{f}^{\top}\mathbf{m}_k/\tau_{\mathrm{csc}}\big)}{\sum_{l=1}^{K}\exp\big(\mathbf{f}^{\top}\mathbf{m}_l/\tau_{\mathrm{csc}}\big)}, \tag{A.3}$$

whose numerator enforces compactness toward the true centroid and whose denominator disperses the centroids, with τ_{csc} the temperature.

Training schedule. We optimize with AdamW at learning rate 10^{-3} and weight decay 10^{-4}, with a batch size of 32 and gradient clipping at norm 5.0. The contrastive term $\mathcal{L}_{\mathrm{csc}}$ is disabled for the first 8 epochs (warm-up), so the network first learns a useful feature space under the primary task

loss before $\mathcal{L}_{\text{csc}}$ starts shaping it. Training runs for up to 400 epochs with early stopping after 40 evaluation cycles without improvement on the held-out split. Per-frame inputs are globally normalized to zero mean and unit variance using statistics from the training split, with light additive Gaussian augmentation (σ=0.01) and temporal masking (drop a random 5% of frames per clip to zero) during training only. Tab. A.2 lists the main hyperparameters.

Table A.2: Main hyperparameters of UNVEIL.

Hyperparameter	**Value**
Input	
Joints (J)	15
Sequence length (T)	256
Frame rate	30 fps
Architecture	
ST layers (N)	10
Base channels	96
Signature-bank prototypes (n_p)	100
Trajectory embedding dim	256
Loss	
$\mathcal{L}_{\text{csc}}$ weight (λ_{csc})	0.1
$\mathcal{L}_{\text{csc}}$ temperature (τ_{csc})	0.125
Contrastive warmup epochs	8
Optimization	
Optimizer	AdamW
Learning rate	10^{-3}
Weight decay	10^{-4}
Batch size	32
Max epochs	400

A.3 Additional Baseline Results

We evaluate four baselines under the same splits and protocol as Tab. 1; results are in Tab. A.3.

MLP. A two-layer MLP (256 and 128 units, ReLU, dropout 0.3) takes the 27 interpretable features of App. A.4 as input. It has no access to joint connectivity or frame-level temporal information.

GaitGraph [33]. GaitGraph is a ResGCN-based network for skeleton-based gait recognition, which we adapt by building its adjacency from the G1 kinematic tree and feeding it the same input as UNVEIL. It tests whether an architecture designed for human gait recognition picks up operator signatures from retargeted trajectories.

SGN [43]. Following Zhang et al. [43], we concatenate each joint's dynamics feature with a learned joint-type embedding. A content-adaptive adjacency, computed per frame by non-local attention over joints, drives three residual GCN layers (128, 256, and 256 channels). Joints are then max-pooled into a frame-level vector with an added frame-index embedding, and two temporal convolutions (256 and 512 channels) produce the sequence representation.

DSGCN [36]. Following Xie et al. [36], we split each spatial graph convolution into three parallel branches over five body parts (two arms, two legs, and the trunk): a node type-aware branch with part-specific projections, an edge type-aware branch with a separate transformation for each of the 15 body-part-pair edge types, and a general adaptive-graph branch. The branch outputs are concatenated and fused by a 1×1 convolution in each of the ten layers (64 base channels, doubled at the 5th and 8th layers). DSGCN is trained with SGD with momentum 0.9 and weight decay 5×10^{-4}, starting from a learning rate of 0.1 that decays by cosine annealing over 100 epochs.

A.4 Interpretable Feature Analysis

Feature extraction. Each demonstration is a Unitree G1 trajectory with 35 channels (6 root degrees of freedom and 29 joint angles) sampled at 120 Hz. We define nine body regions: root translation,

Table A.3: Operator attribute inference using four baseline architectures adapted to our pipeline. Full protocol matches Tab. 1. MLP and GaitGraph use no spatiotemporal graph convolution specific to our design; SGN and DSGCN replace UNVEIL's spatial graph with well-tested alternatives.

	Task → / Eval. Setting ↓	Gender Acc(%)↑	Weight MedAE↓	Height MedAE↓	Age MedAE↓	Re-ID Top-1 | Top-5 Acc(%)↑	Linkage Acc(%)↑
MLP	Unseen operators	79.4	7.4 kg	5.1 cm	3.2 yr	—	—
	Unseen demos of seen operators	87.6	4.9 kg	2.8 cm	2.1 yr	85.3 | 95.2	81.7
GaitGraph [33]	Unseen operators	74.8	8.7 kg	6.1 cm	3.3 yr	—	—
	Unseen demos of seen operators	83.4	6.9 kg	5.3 cm	2.7 yr	88.4 | 97.1	87.6
SGN [43]	Unseen operators	78.2	6.9 kg	4.7 cm	2.5 yr	—	—
	Unseen demos of seen operators	81.4	3.0 kg	3.1 cm	2.2 yr	79.0 | 90.0	80.1
DSGCN [36]	Unseen operators	80.0	7.1 kg	4.4 cm	2.2 yr	—	—
	Unseen demos of seen operators	82.8	4.8 kg	2.8 cm	1.7 yr	95.9 | 98.0	91.1

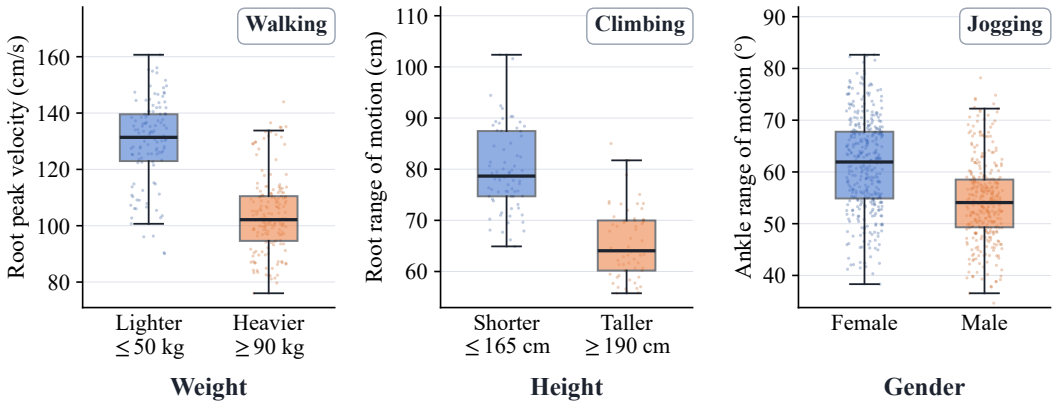


Figure A.2: Distributions of strongly correlated features for weight (root peak velocity, walking), height (root range of motion, climbing), and gender (ankle range of motion, jogging). Operators are grouped as lighter ($\leq$50 kg) vs. heavier ($\geq$90 kg), shorter ($\leq$165 cm) vs. taller ($\geq$190 cm), and female vs. male; each dot is one demonstration.

root rotation, hip, knee, ankle, waist, shoulder, elbow, and wrist, merging the left and right channels of bilateral joints. For each region we compute the mean angular velocity, the range of motion (maximum minus minimum angle), and the peak angular velocity, aggregating multi-axis regions with the ℓ_2 norm, which gives 27 features per demonstration.

Correlation analysis. We keep activities with at least 10 demonstrations and 5 operators and, for each (activity, feature, attribute) triple, compute Pearson's r across the operators who performed the activity, with gender encoded as 0 for female and 1 for male. Pairs with fewer than five valid observations or near-zero variance are discarded. Since the four attributes are correlated with one another, we also compute partial correlations. For a feature f and an attribute A, with C the remaining three attributes, we regress both f and A on $[\mathbf{1}, C]$ by ordinary least squares and take Pearson's r between the two residuals as r_{partial}. Partial correlations are computed only for activities with at least 30 operators. Tab. 3 in the main text reports the mean raw and partial correlations of the most strongly correlated features for each attribute. Fig. A.2 shows, for weight, height, and gender, the distribution of one of these features within a single activity, comparing operators at the two ends of each attribute.

A.5 Details on Anonymization

Method. A motion encoder E_M and a privacy encoder E_P split a trajectory into motion content and operator signature, and a shared decoder D reconstructs the trajectory from both. To anonymize a source trajectory $\mathbf{X}$, we replace its privacy embedding with that of a target trajectory: $\tilde{\mathbf{X}} = D(E_M(\mathbf{X}), E_P(\mathbf{X}_{\text{target}}))$, where $\mathbf{X}_{\text{target}}$ is the training trajectory whose operator is most distant from the source operator in standardized attribute space (age, gender, height, weight). We train

Table A.4: Physics-based utility evaluation of anonymized trajectories. *Top:* SONIC whole-body tracking accuracy on the six highest-volume BONES-SEED categories (frozen checkpoint, Isaac Lab, randomized dynamics). *Middle:* kinematic smoothness (lower is smoother for acceleration and jerk; lower is better for self-collision). *Bottom:* behavioral cloning success on operators in the bottom/top 5% tails of age, height, and weight, averaged over the three attributes (per-attribute breakdown in Tab. A.5).

	Original	Anonymized	Δ (pp)
SONIC tracking accuracy (%)↑			
Gestures	98.8	93.4	−5.4
Basic locomotion styles	99.2	86.6	−12.5
Object interaction	93.7	75.4	−18.3
Baseline	71.4	52.9	−18.5
Basic locomotion neutral	91.1	70.3	−20.8
Object manipulation	96.4	55.4	−41.1
Overall	91.8	72.3	−19.4
	Original	Anonymized	Ratio
Kinematic smoothness			
Mean ang. accel. (rad/s^2)↓	10.4	5.2	0.50×
Mean ang. jerk (rad/s^3)↓	836	278	0.33×
Self-collision (% frames)↓	17.6	1.6	0.09×
	Original	Anonymized	Δ (pp)
Policy learning: tail success (%)↑			
BC-RNN	33.6	35.9	+2.3
BC-Transformer	60.8	62.7	+1.9

$\{E_M, E_P, D\}$ adversarially against one instance of UNVEIL per attribute, denoted f_a:

$$\mathcal{L}_{\text{anon}} = \left\| D(E_M(\mathbf{X}), E_P(\mathbf{X})) - \mathbf{X} \right\|_2^2 + \lambda \sum_a \Big[\mathcal{L}_a\big(f_a(E_P(\mathbf{X})), y_a\big) - \mathcal{L}_a\big(f_a(E_M(\mathbf{X})), y_a\big) \Big], \tag{A.4}$$

where y_a is the ground-truth value of attribute a and $\mathcal{L}_a$ is cross-entropy for gender and re-identification and MSE for age, height, and weight. Each f_a is updated alternately to recover its attribute from both factors, while the encoders learn to concentrate the operator signature in E_P and remove it from E_M.

Physics-based tracking and motion quality (Tab. A.4). We evaluate whether a frozen SONIC [21] checkpoint can track anonymized references in Isaac Lab with randomized friction and link masses. A clip succeeds if the robot tracks the reference to completion without exceeding SONIC's termination thresholds. Overall success drops from 91.8% to 72.3%, with the drop varying across activity categories. Anonymized trajectories also have lower angular acceleration and jerk and fewer self-collisions than the originals.

Policy equity across the operator distribution (Tab. A.5). We train BC-RNN and BC-Transformer from robomimic [22] separately on original and anonymized trajectories. The policy observes the robot's current joint angles and predicts the joint-angle change for the next frame; at evaluation it starts from an unseen operator's initial pose and is rolled out autoregressively, and a rollout succeeds if it stays within a tolerance of the reference. We compare success on operators in the middle 90% of each attribute with those in the bottom and top 5%. Anonymization narrows this gap for every attribute and both architectures, for example from +9.9 to +2.4 pp for height with BC-Transformer. A smaller gap does not imply uniform improvements: BC-Transformer weight-tail success decreases from 65.1% to 63.7%, and middle-group success decreases for every attribute and both architectures.

Transfer to other architectures (Tab. A.6). To test whether the privacy reduction is specific to UNVEIL, we also train the anonymizer against SGN [43] and DSGCN [36] and evaluate all nine combinations of training adversary and evaluator. Re-ID Top-1 stays between 16.8% and 21.4% in

Table A.5: Behavioral cloning equity across the operator distribution. Success rate (%) for operators in the middle 90% vs. the bottom/top 5% tails of each attribute. *Gap* = middle − mean of tails; lower is more equitable.

		Age			Height			Weight		
Architecture	Trained on	Tails	Mid	Gap	Tails	Mid	Gap	Tails	Mid	Gap
BC-RNN	original	30.0	49.6	+19.6	35.1	45.8	+10.7	35.7	45.3	+9.6
	anonymized	32.4	43.5	+11.1	36.4	41.5	+5.1	38.9	40.5	+1.6
BC-Trans.	original	56.6	74.0	+17.4	60.8	70.7	+9.9	65.1	69.2	+4.1
	anonymized	59.1	70.4	+11.3	65.2	67.6	+2.4	63.7	66.4	+2.7

Tails = mean of bottom-5% and top-5% success rates.

Table A.6: Cross-architecture anonymization: full evaluation grid. Each row trains the anonymizer against one architecture; each column block evaluates with a (possibly different) architecture. Diagonal entries (shaded) are matched; off-diagonal entries test transfer. Re-ID Top-1 on original trajectories: 97.2%.

Trained against	Evaluated by	Operator Attributes						Utility	
		Re-ID Top-1↓	Linkage Acc↓	Gender Acc↓	Age MedAE↑	Height MedAE↑	Weight MedAE↑	Action Rec. Top-1↑	Top-5↑
UNVEIL	UNVEIL	16.8%	53.1%	51.8%	5.4 yr	7.7 cm	8.5 kg	80.0%	98.1%
	SGN	18.2%	58.1%	58.8%	7.4 yr	8.1 cm	9.6 kg	76.9%	97.1%
	DSGCN	18.7%	59.1%	57.3%	7.0 yr	8.2 cm	9.6 kg	77.4%	98.0%
SGN	UNVEIL	19.2%	60.2%	57.1%	6.3 yr	8.2 cm	9.4 kg	78.2%	94.6%
	SGN	17.0%	58.3%	54.9%	6.4 yr	8.2 cm	10.2 kg	79.6%	94.7%
	DSGCN	19.4%	58.9%	55.8%	6.3 yr	8.0 cm	10.0 kg	77.4%	93.1%
DSGCN	UNVEIL	21.4%	55.3%	52.9%	7.1 yr	7.5 cm	8.7 kg	80.8%	96.8%
	SGN	20.1%	54.2%	53.8%	7.0 yr	7.2 cm	8.5 kg	80.9%	96.3%
	DSGCN	17.6%	54.0%	53.1%	7.6 yr	8.0 cm	9.7 kg	81.4%	96.8%

every combination, compared with 97.2% on original trajectories, and action recognition Top-1 stays between 76.9% and 81.4%.

A.6 Operator Survey Protocol

We conducted an anonymous online survey on Prolific. Participants first read a scenario in which their everyday activities are recorded, mapped onto a robot body, and the resulting robot motion recording is shared publicly for robot learning research; they then answered Q1–Q5, read a disclosure that retargeted robot recordings can reveal a person's age, gender, height, weight, and identity, and answered Q6–Q9. The survey also included one attention-check item.

Q1. How willing would you be to take part in a program like this, contributing recordings of your everyday activities, where your robot motion recording is then shared publicly? *(Extremely willing / Somewhat willing / Unsure / Somewhat unwilling / Not willing at all)*

Q2. Please describe the reason for your answer above. *(free text)*

Q3. What do you think can be learned from a shared robot motion recording? *(free text)*

Q4. How likely is it that a robot motion recording could reveal a person's personal characteristics, such as their age, gender, height, or weight? *(Very likely / Likely / Unsure / Unlikely / Very unlikely)*

Q5. In the scenario, your movements are changed to fit the robot's body before the recording is shared. How much do you think this hides who you are? *(Completely / A lot / Somewhat / A little / Not at all)*

Q6. Knowing what you just read, how willing would you be to take part in a program like this, contributing recordings of your everyday activities, where your robot motion recording is then shared publicly? *(same scale as Q1)*

Q7. Please describe the reason for your answer above. *(free text)*

Q8. How would each of the following change your willingness to take part in the program? *(Much more willing / Somewhat more willing / No change / Somewhat less willing / Much less willing)*

- Being paid more
- Being able to have your data deleted later
- Being told each time a company or researcher uses your data
- Knowing your identity and personal characteristics cannot be recovered from the recording
- Being able to limit who can access the data (for example, approved researchers only)
- Being able to choose what your data is used for (for example, academic research only)
- A guarantee that your data will not be sold or shared with other companies

Q9. Is there anything else (not listed above) that would make you more or less willing to take part? *(free text)*

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction summarize the paper's contributions, assumptions, and evaluation scope. The claims are supported by the experimental results discussed in the paper.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses its limitations in the conclusion section.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: Showing proof is not applicable to this paper.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper discloses the methodology and hyperparameter values needed to reproduce the results with the provided code.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We release our source code with the hyperparameters in App. A.2. BONES-SEED is publicly available from Bones Studio under the BONES-SEED Dataset License Agreement; we do not redistribute it.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: The paper mentions all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) in the App. A.2.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: The paper reports the correlation analysis for the statistically significant features.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper mentions in App. A.2 the machine and GPU used for training and how long the experiments took.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The paper has maintained anonymity and followed all other guidelines mentioned in the Ethics Guidelines.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper discusses the work's impact on the robotics community.

    Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper only released the scripts that use an existing dataset to generate the results.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We use Motion Data by Bones Studio (`https://bones.studio/`) under the BONES-SEED Dataset License Agreement and credit it in the Acknowledgments as per the license requirements. Our re-identification and linkage tasks operate only within the dataset. They test whether two robot trajectories came from the same pseudonymous operator label (included in the dataset metadata), and we do not make any attempt to map a trajectory, label, or attribute to a real person, subject, or actor. We cite the retargeting methods and baselines we use.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.

- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: No new assets are introduced.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: Sec. 4.5 describes the Prolific survey procedure; participants were paid $2.00 for an approximately five-minute survey. The survey questionnaire is included in App. A.6.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [Yes]

Justification: The survey in Sec. 4.5 was an anonymous online questionnaire, distributed through Prolific about participants' attitudes toward teleoperation data collection. It collected no identifying information and posed no more than minimal risk. The study was reviewed by UCI's Institutional Review Board (IRB) and determined to be exempt. Participants were informed of the purpose of the survey before taking part and could withdraw at any time.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLM use was limited to paper formatting and writing, so no declarations are needed.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.