

MorphMimic: Morphology-Conditioned World Model and Policy Sharing Across Humanoids

Lizhi Yang¹, Fengze Xie^{1,*}, Yuyang Wang^{1,*}, Lu Gan², Yisong Yue¹, and Aaron D. Ames¹

Abstract—As more humanoid platforms become available, training a separate whole-body motion tracker for each robot becomes costly. We present MorphMimic, which trains one world model and one policy that takes the embodiment as input. A canonical joint and body layout maps the Unitree G1, AgiBot X2 Ultra, and Booster T2 to a common coordinate system. A physical descriptor of actuation, geometry, and mass conditions the world model, policy decoder, and critics based on the robot being controlled. An alignment loss brings the latent commands for corresponding motions closer across robots, while a second encoder learns to reproduce those commands from pelvis, ankle, and wrist targets through the same decoder. Both interfaces use position-free targets with joint-encoder and inertial orientation feedback. Across four sharing configurations evaluated on 128,786 common clips, every shared configuration stays within 0.02 mean return of a single-robot specialist on all six robot-interface pairs. On a physical Unitree G1, the same unified policy performs dance, walking, and real-time teleoperation. Code and video are available at <https://lzyang2000.github.io/morphmimic>.

Index Terms—Humanoid robot systems, reinforcement learning, transfer learning, whole-body motion planning and control.

I. INTRODUCTION

WHOLE-BODY motion tracking converts human reference motion into humanoid behavior and supports teleoperation and command interfaces built on a tracking policy [1], [2]. More humanoid platforms are now available, and many have similar sizes and kinematic layouts. Whole-body trackers are usually trained for one robot [1], [2], so a new embodiment requires retraining or adaptation [3]. A controller that takes the embodiment as an input would let a new humanoid use the motion repertoire, training procedure, and command interfaces developed for the others. Experience collected on one robot could then benefit the rest.

Sharing a controller across humanoids is difficult because similar kinematic structures can have different physics. Joint axes, sign conventions, neutral poses, limb lengths, proportional-derivative (PD) gains, and torque limits differ, so the same action produces different motions on different bodies. The problem is harder when the policy learns through a world model. SuperMimic [2] trains a tracker by differentiating through short rollouts of a learned dynamics model. Prediction errors therefore affect both the imagined trajectory and the policy gradient, and a shared model must predict each robot well enough for that gradient to remain useful. Prior cross-embodiment methods condition policies on

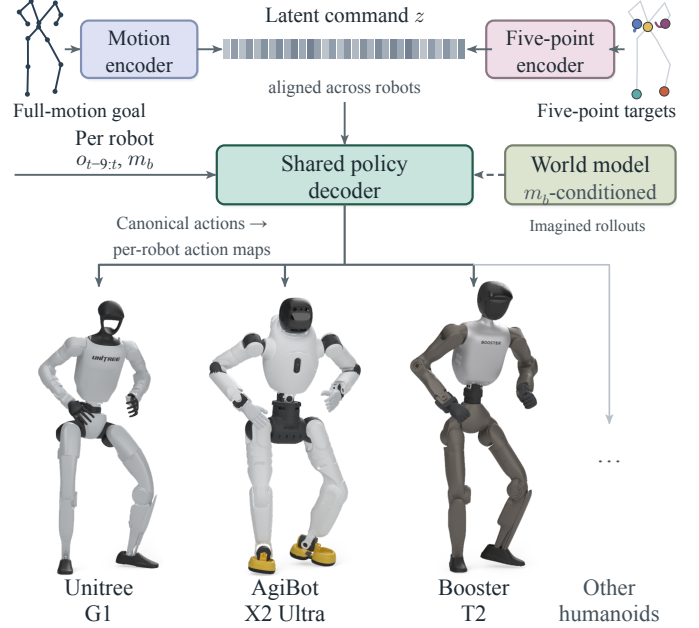


Figure 1. One world model and one policy shared across humanoids. Full-motion latents are aligned across robots, and a five-point encoder learns to match them through the same decoder. Morphology conditions the world model and decoder; the dashed arrow marks imagined rollouts. Robots are simulator renders at a common scale; latent cells are schematic.

hardware descriptions [4], share locomotion policies through morphology-aware encoding or semantic joint alignment [5]–[7], condition world models on morphology across robot bodies [8], [9], or transfer and co-train humanoid trackers with robot-specific modules [3], [10]. We address this setting by sharing one differentiable world model and one complete tracking policy across humanoids.

MorphMimic treats the embodiment as an input to one world model and one policy (Fig. 1). A canonical layout maps joints by kinematic function, with joint-specific signs and offsets, to a common joint order and set of tracked bodies. A physical descriptor of actuation, geometry, and mass conditions the world model, critics, and policy decoder. Network weights are shared, while normalization statistics, reference motions, and replay data remain robot-specific. A new humanoid that fits the layout can therefore be added with a joint map, a descriptor, and retargeted motions. An alignment loss pulls the latent commands for corresponding retargeted frames toward their cross-robot mean. A five-point encoder for pelvis, ankle, and wrist targets learns to reproduce the full-motion latents of its robot and drives every robot through the shared decoder. Other interfaces can target the same latent

*Equal contribution. ¹California Institute of Technology, CA 91106, USA.

²Georgia Institute of Technology, Atlanta, GA 30332, USA. Corresponding author email: lzyang@caltech.edu. This research is funded by The Dow Chemical Company project #227027AW.

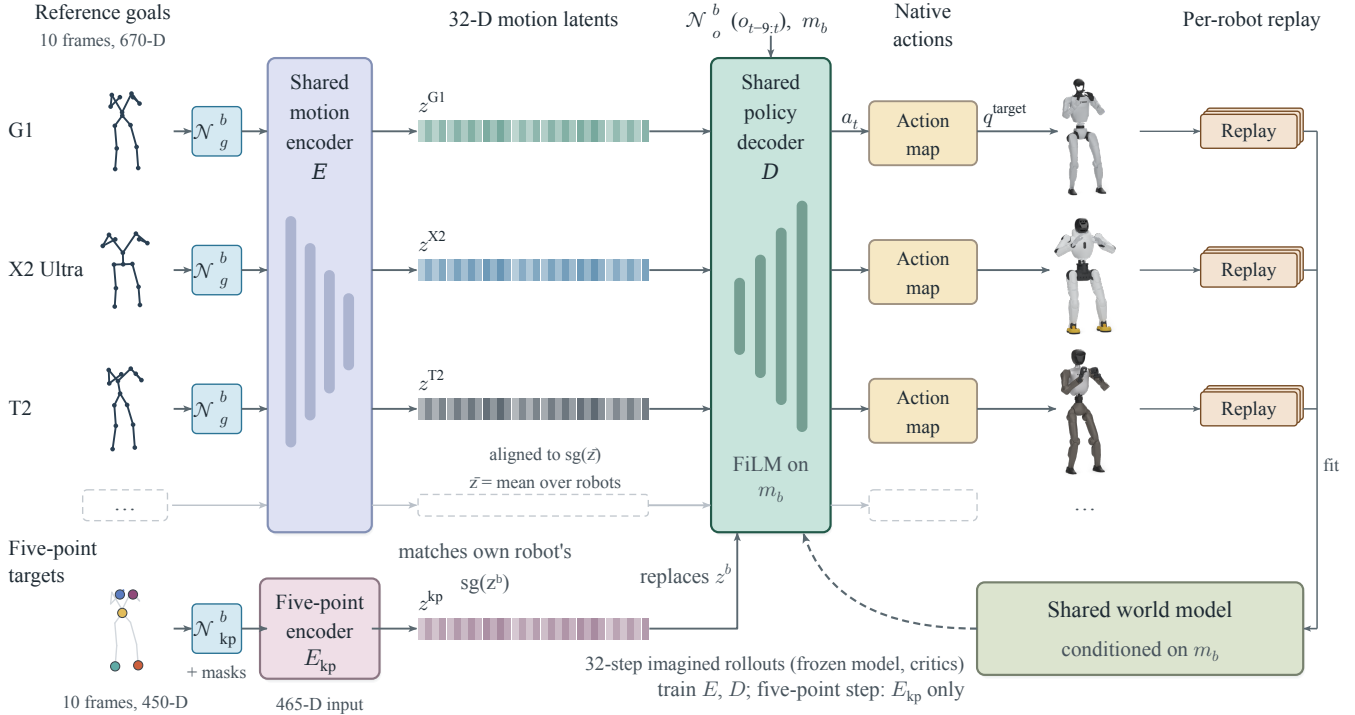


Figure 2. Shared policy and command interfaces. Robot-specific normalizers feed a shared motion encoder, whose 32-dimensional latent enters the shared decoder with proprioception and morphology; morphology also modulates decoder features through FiLM. Motion latents of corresponding frames align to their mean, and the five-point encoder matches its robot’s motion latent. Latent bars are schematic.

space. Both interfaces use position-free targets with feedback from joint encoders and the inertial measurement unit (IMU).

Across four sharing configurations evaluated on the 128,786 clips common to the three training libraries, every shared configuration stays within 0.02 mean return of a single-robot specialist on all six robot–interface pairs. The best completion rate varies by robot and interface, while the five-point interface lowers mean return across configurations. The unified run uses 16.3 GPU-hours, compared with 13.9 GPU-hours for the three specialists together, and produces one set of weights for all robots. On a physical humanoid, the same weights perform dance, walking, and real-time teleoperation through a VR headset and ankle trackers (Fig. 5).

Our contributions are a common robot representation and morphology-conditioned training for one differentiable world model and one whole-body tracking policy across humanoids with different joint conventions, geometry, and actuation. We also introduce cross-robot alignment of full-motion latent commands and a five-point interface that uses the shared decoder, both with position-free targets. A full-corpus simulation compares the shared policy with single-robot specialists through both command interfaces, and hardware experiments on a physical humanoid evaluate dance, walking, and teleoperation with the shared policy.

II. RELATED WORK

A. Cross-Embodiment Control Policies

Sharing control policies across robot bodies has a long history. Chen et al. [4] condition a universal policy on a

vector encoding the robot hardware. Shared Modular Policies [11] reuse an actuator-level controller across bodies and coordinate modules through message passing. MetaMorph [12] trains morphology-conditioned Transformers across modular robot designs and transfers them to unseen morphologies. For dynamic legged locomotion, GenLoco [13] combines motion imitation with procedurally generated quadruped morphologies and transfers to unseen simulated and real robots, using sensor and action histories to infer the body. URMA [5] shares a locomotion policy across embodiments with different joint configurations through attention-based observation encoding and joint-conditioned action decoding. Bohlinger et al. [6] extend this idea with extreme Embodiment Randomization across 50 legged robots, including zero-shot transfer to unseen humanoids and quadrupeds. XHugWBC [7] combines semantic joint alignment with physics-consistent morphological randomization for cross-humanoid locomotion. MS-PPO [14] encodes the permutations and sign flips induced by the morphological symmetry of a single robot in a graph policy, using a symmetry-equivariant actor and symmetry-invariant critic for locomotion on the Unitree Go2 and G1. We apply explicit morphology conditioning to whole-body tracking with a fixed semantic joint layout and a decoder conditioned on geometry and actuation. The same physical descriptor also conditions the learned dynamics.

B. Cross-Embodiment Whole-Body Tracking

Whole-body tracking across bodies also needs a command representation that relates motions between robots.

Any2Any [3] transfers a pretrained humanoid tracker to a new body through kinematic alignment and parameter-efficient adaptation of dynamics-sensitive policy modules. X-WBC [10] jointly trains a shared temporal backbone with robot-specific command encoders, proprioception encoders, and action decoders. It aligns command tokens from robot, human, and sparse motion inputs, separating the requested motion from its execution by a particular robot. In MorphMimic, the goal encoders and action decoder are shared across robots. Corresponding retargeted frames supervise alignment of motion latents, and the five-point encoder learns to reproduce each robot’s full-motion latent so sparse targets can use the shared decoder.

C. World Models for Policy Learning

World models offer another way to share information across bodies. PWM [15] learns policies through differentiable multi-task world models. MS-HGNN [16] incorporates kinematic structure and morphological symmetry into a heterogeneous graph network for robot dynamics. QWM [8] conditions a recurrent latent world model on physical morphology and learns quadrupedal locomotion policies through imagined rollouts. WestWorld [9] scales a trajectory world model across diverse robotic systems with a system-aware mixture of experts and a morphology-based structural embedding. SuperMimic [2] applies differentiable world-model learning to humanoid whole-body tracking and supports additional command encoders. MorphMimic extends this setting from one body to several humanoids by sharing the state-space dynamics model and tracking policy through a common representation, morphology conditioning, and command alignment. Both command interfaces use one morphology-conditioned action decoder.

III. METHODOLOGY

MorphMimic follows SuperMimic’s alternating loop of simulator data collection, world-model fitting, and policy updates through imagined rollouts [2, Sections 3.1–3.3]. Fig. 2 shows how the loop is shared across robots. We first define the tracking problem and a common representation for the robots. We then describe the shared world model, the morphology-conditioned policy with aligned latent commands, the five-point interface, and multi-robot training. Robot-specific details and all constants are given in Appendix A.

A. Problem Formulation

Let $b \in \mathcal{B} = \{\text{G1}, \text{X2}, \text{T2}\}$ denote the embodiment, s_t^b its state, and $a_t \in \mathbb{R}^{N_j}$ the canonical action over N_j body joints. The simulator transition is $s_{t+1}^b \sim P_b(\cdot | s_t^b, a_t)$ with control interval Δt . A reference motion supplies a target state $\bar{s}_t^b$ and a goal g_t^b constructed from a window of K target frames, and the policy uses a history $o_{t-K+1:t}^b$ of the last K proprioceptive observations. With the discount factor γ , we seek shared policy parameters θ that maximize the equally weighted tracking objective

$$J(\theta) = \frac{1}{|\mathcal{B}|} \sum_{b \in \mathcal{B}} \mathbb{E} \left[\sum_{t=0}^{T_b-1} \gamma^t r_{t+1}^b \right], \quad (1)$$

where T_b is the episode length and r_{t+1}^b evaluates the state reached by action a_t against the target $\bar{s}_t^b$.

We use SuperMimic’s motion-tracking reward and weights [2, Appendix A, Table A.2], evaluated on the N_b canonical body slots of Section III-B with the torso-equivalent body as the tracking anchor, and denote the maximum reward per step by $r_{\max}$. An episode terminates when the anchor-height error e_z exceeds $\bar{e}_z$ or the tilt error e_u exceeds $\bar{e}_u$, where the tilt error is the absolute difference between the vertical gravity components in the robot and reference anchor frames. Reaching the end of the reference counts as a time-out, and all three robots share these tolerances. Table II in Appendix A summarizes the simulated bodies.

B. Common Robot Representation

1) *Canonical Joints and Actuator Targets*: We map joints by kinematic function. For canonical slot j , let $p_b(j)$ be the corresponding native joint index, $\epsilon_{b,j} \in \{-1, 1\}$ its sign, and $\delta_{b,j}$ its angular offset. A native joint position becomes

$$q_j^c = \epsilon_{b,j} (q_{p_b(j)}^b - \delta_{b,j}). \quad (2)$$

Velocities and other joint differences use the permutation and sign without the offset. This map aligns joint slots across the naming, sign, and neutral-pose conventions of the robots. The learned networks account for the remaining kinematic differences. With the effort limit $\tau_{b,j}^{\max}$, the proportional gain $k_{p,b,j}$, and an action scale κ_a , the action is a scaled displacement from the robot’s default pose, with native position target

$$q_{p_b(j)}^{b,\text{target}} = q_{p_b(j)}^{b,\text{default}} + \epsilon_{b,j} \alpha_{b,j} a_{t,j}, \quad \alpha_{b,j} = \frac{\kappa_a \tau_{b,j}^{\max}}{k_{p,b,j}}. \quad (3)$$

Since $k_{p,b,j} \alpha_{b,j} = \kappa_a \tau_{b,j}^{\max}$, a unit action changes the proportional torque at a fixed joint state by the fraction κ_a of the effort limit. The displacement in (3) is measured from the native default pose, so the offset in (2) affects only the absolute joint configuration.

2) *Tracked Bodies and State*: The N_b body slots contain a pelvis equivalent, three bodies on each leg, a torso equivalent, and three bodies on each arm. The pelvis and torso occupy slots 0 and 7. Gravity and angular velocity are measured at each robot’s IMU body. The world-frame state contains joint positions and velocities, together with the positions, quaternion orientations, and linear and angular velocities of the tracked bodies. Let $p_{P,t}$ denote the world-frame position of the pelvis equivalent, and let F_t be the frame with origin $(p_{P,t,x}, p_{P,t,y}, 0)$ and the full orientation of the pelvis equivalent. The transform Φ_{F_t} expresses body positions relative to this origin and rotates positions, orientations, and velocities into F_t , while leaving canonical joint coordinates unchanged. Each orientation is represented by the first two columns of its rotation matrix. A separate context $\xi_b \in \mathbb{R}^{d_\xi}$ contains a foot-friction coefficient, joint encoder biases, and a center-of-mass offset.

3) *Observations and Position-Free Goals*: The policy uses SuperMimic’s proprioceptive observation [2, Appendix A, Table A.1] over a K -frame history. Joint positions, measured relative to the default pose and including encoder bias, and joint velocities are expressed in canonical coordinates. Angular

velocity and projected gravity are expressed in the IMU-body frame, and the previous canonical action completes the observation. The history is normalized by a robot-specific transform $\mathcal{N}_o^b$.

The full-motion goal concatenates K target frames, each holding the reference joint positions and velocities and the position and six-dimensional orientation of the torso anchor (slot 7). The frames are the current reference frame and the $K - 1$ subsequent frames at the control rate; the final frame is repeated at the clip boundary. Anchor positions are expressed in the reference pelvis heading frame at the first goal frame. Its origin is the horizontal position of the reference pelvis at zero height. This frame describes the path ahead and the absolute target height without depending on the robot’s position. Anchor orientations are expressed relative to the robot’s current torso anchor and provide orientation feedback, including yaw. The goal is normalized by $\mathcal{N}_g^b$. With these position-free goals, the policy inputs are the reference targets, joint-encoder measurements, angular velocity, and an IMU-based attitude estimate. All of these inputs are available from onboard sensing and the reference, which simplifies deployment.

4) *Morphology Descriptor*: The static descriptor $m_b \in \mathbb{R}^{d_m}$ of dimension d_m is

$$m_b = \mathcal{S}([\log \alpha_b, \log \mathbf{k}_{p,b}, \log \mathbf{k}_{d,b}, \log \tau_b^{\max}, \eta_b]), \quad (4)$$

where $\mathbf{k}_{p,b}$ and $\mathbf{k}_{d,b}$ are the proportional and derivative gains, the four actuator blocks have N_j entries in canonical joint order, and η_b holds geometric and mass measurements, namely thigh, shank, upper-arm, and forearm lengths, hip width, torso height, total mass, and leg and arm subtree masses. The transform $\mathcal{S}$ scales the features to $[-1, 1]$ across the three robots and maps constant features to zero.

C. Reference Motions

We use BONES-SEED [17]¹ with its provided G1 retargets and retarget the same source motions to X2 and T2 with NVIDIA’s SOMA Retargeter [18]. Because all robots share one human-motion source, every clip has corresponding reference frames on the three bodies, which the latent alignment of Section III-E relies on. The retargeted motions are converted to reference states at the control rate within the soft joint limits of the simulator, as described in Appendix A.

D. Shared World Model

Shared training uses SuperMimic’s H_m -step autoregressive model fitting [2, Section 3.2], with one dynamics network and robot-specific replay buffers. The network predicts a residual in the frame F_t . With $u_t = \Phi_{F_t}(s_t)$ and a diagonal context scaling S_ξ that brings friction, encoder bias, and center-of-mass offset to comparable magnitudes, the update is

$$\begin{aligned} x_t^b &= [\mathcal{N}_s^b(u_t), e_t^b, a_t, S_\xi \xi_b, m_b], \\ \hat{s}_{t+1}^b &= \Phi_{F_t}^{-1}(u_t + \sigma_{\Delta,b} \odot f_\omega(x_t^b)). \end{aligned} \quad (5)$$

Here, $\mathcal{N}_s^b$ uses robot-specific motion-library statistics, e_t^b contains gravity in F_t and the normalized world-frame heights of

the tracked bodies, and the fixed scale $\sigma_{\Delta,b}$ is the root-mean-square (RMS) one-step residual in warm-up replay. Because the residual output layer is initialized to zero, the initial model predicts an unchanged state in F_t .

The state-fitting term compares predicted and recorded features in the frame of the recorded state. It sums squared errors after division by $\sigma_{\Delta,b}$ and averages them over the H_m steps and the sampled windows. A termination head is fit with binary cross-entropy on the same windows, which gives $\mathcal{L}_{\text{wm}}^b = \mathcal{L}_{\text{state}}^b + \mathcal{L}_{\text{term}}^b$, as in the analytic reward and observation case of SuperMimic’s Eq. (1). With learned positive gains k_z and k_u and learned biases b_z and b_u , we model termination as a monotone hazard in the anchor-height and tilt errors:

$$\begin{aligned} p_z &= \text{sigmoid}(k_z(e_z - \bar{e}_z) + b_z), \\ p_u &= \text{sigmoid}(k_u(e_u - \bar{e}_u) + b_u), \\ \hat{d} &= 1 - (1 - p_z)(1 - p_u), \end{aligned} \quad (6)$$

with the termination thresholds of the simulator.

E. Morphology-Conditioned Policy with Aligned Commands

A motion encoder maps the normalized goal to a latent command of dimension d_z , and a decoder maps this latent, the proprioceptive history, and the morphology descriptor to an action:

$$z_t^b = E(\mathcal{N}_g^b(g_t^b)), \quad a_t = D(z_t^b, \mathcal{N}_o^b(o_{t-K+1:t}^b), m_b). \quad (7)$$

The encoder and decoder follow SuperMimic’s architecture [2, Table A.4], with $z_t^b \in (-1, 1)^{d_z}$. While the encoder reads only the normalized reference goal, the full morphology descriptor is appended to the decoder and critic inputs. In addition, the decoder uses feature-wise linear modulation (FiLM) [19]. With learned matrices Γ_ℓ and B_ℓ and learned biases μ_ℓ and ν_ℓ , the pre-activation h_ℓ of hidden layer ℓ becomes

$$\tilde{h}_\ell = h_\ell \odot (1 + \Gamma_\ell m_b + \mu_\ell) + B_\ell m_b + \nu_\ell. \quad (8)$$

These modulation parameters start at zero, and training therefore begins from the concatenation-only decoder. The world model receives morphology by concatenation.

1) *Imagined Policy Learning*: Let $\hat{J}_b$ denote SuperMimic’s survival-weighted imagined return [2, Eq. (2)], computed over H_π -step imagined rollouts with the dynamics and critic conditioned on m_b . The policy gradient passes through the frozen model and critic, goals are recomputed from the predicted states and the reference window, and critic bootstrapping is retained at time-outs. Because the reward depends on the displacement of the robot from the reference, both critics also receive the reference anchor positions expressed in the current anchor frame of the robot for all K goal frames, divided by the width σ_p of the position-reward kernel. These inputs are used only during training and remain outside the position-free policy inputs. Critics fit detached λ -return targets following SuperMimic’s Eq. (3).

2) *Latent Command Alignment*: Corresponding reference frames should produce similar latent commands on all robots. At each policy update, the robots draw the same N_a clip-frame pairs from the intersection of their motion subsets, with frame indices restricted to the shortest version of the

¹Motion Data by Bones Studio - <https://bones.studio/>

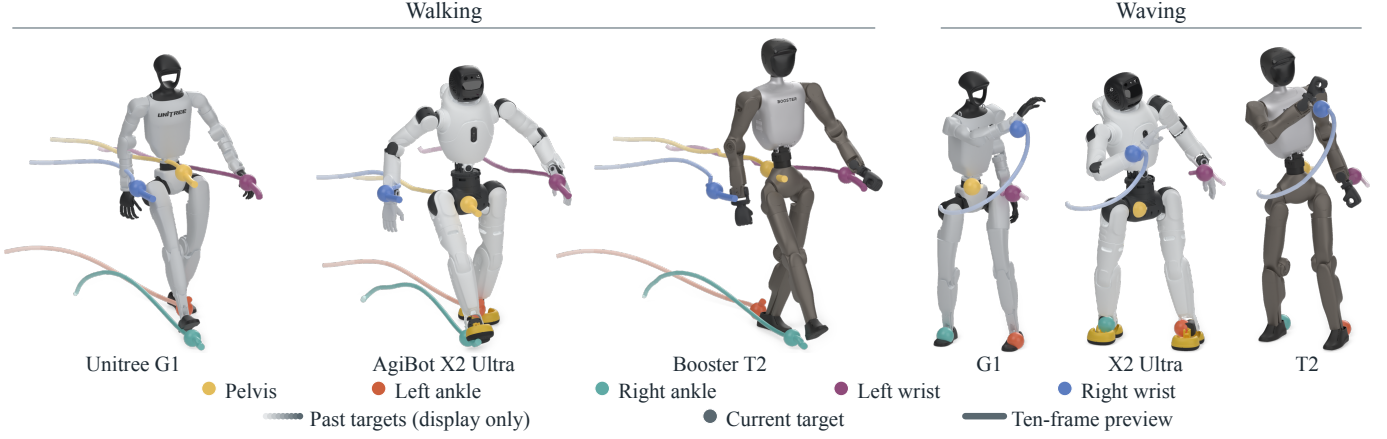


Figure 3. Five-point tracking with the shared policy during the walking (left) and waving (right) phases of one reference motion. Spheres mark the current pelvis, ankle, and wrist targets, solid curves the ten-frame preview, and fading trails the past targets (display only).

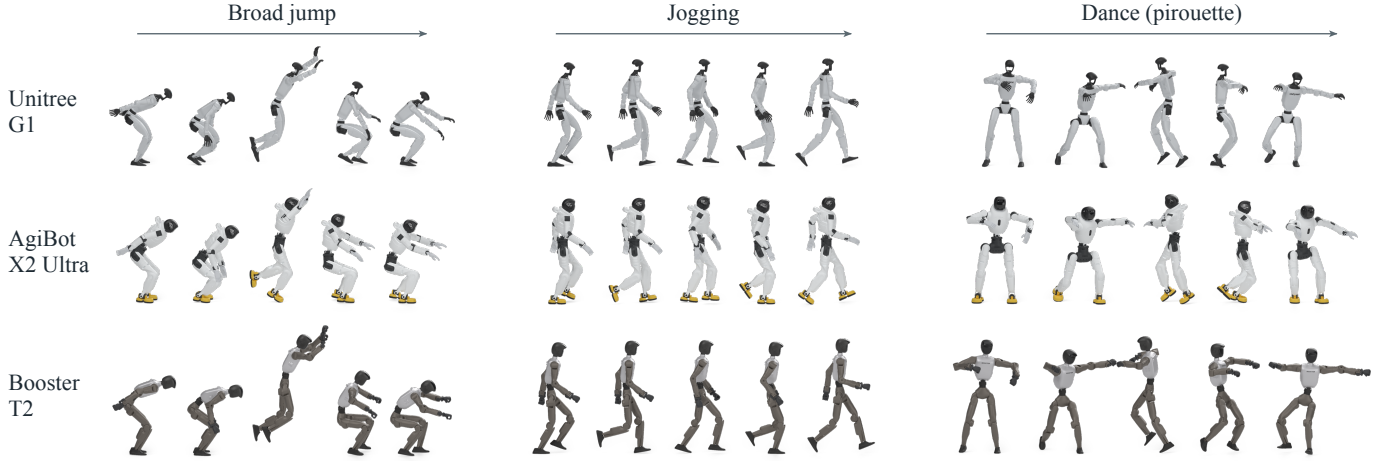


Figure 4. Broad jump, jogging, and dance (pirouette) with the same shared full-motion policy on G1, X2 Ultra, and T2. Each strip shows five simulator poses in time order from left to right; headings and horizontal spacing are adjusted for display.

clip. On each robot, the shared encoder processes that robot’s paired reference goals, computed with the robot placed at the reference state. For pair i , we form the mean latent $\bar{z}_i = |\mathcal{B}|^{-1} \sum_b z_i^b$ and add

$$\mathcal{L}_{\text{align}}^b = \frac{1}{N_a} \sum_{i=1}^{N_a} \|z_i^b - \text{sg}(\bar{z}_i)\|_2^2 \quad (9)$$

to the policy loss with weight β_a . The operator sg stops gradients through the gathered mean, so each robot receives gradients only through its own encoder output. The complete motion-policy loss is $\mathcal{L}_{\text{motion}}^b = -\hat{J}_b + \beta_a \mathcal{L}_{\text{align}}^b$, optimized over the motion encoder and decoder. Following the separation of command and execution in X-WBC [10], the latent describes the requested motion and the morphology-conditioned decoder realizes it on the robot being controlled.

F. Five-Point Command Interface

Based on the multi-interface design of SuperMimic [2, Appendix B], a second goal encoder accepts sparse body-pose targets for the pelvis, the left and right ankles, and the left and right wrists, corresponding to canonical body slots

(0, 3, 6, 10, 13). For the current reference frame and the next $K - 1$ frames, each point contributes a position in meters and a six-dimensional rotation. Target positions use the reference heading frame of the full-motion goal, so that target heights are absolute and target positions are independent of the position of the robot. Target orientations are expressed relative to the full current pelvis rotation of the robot, which provides feedback about roll, pitch, and yaw from onboard sensing.

A running normalizer acts on the target features before masking. A five-bit point mask and a K -bit frame mask identify visible targets. Masked frame–point blocks are zeroed and the mask bits are appended, producing the input $\tilde{g}_t^{b,\text{kp}}$. In the dedicated five-point mode used here, all points and all K frames are visible. The keypoint encoder is a multilayer perceptron with a d_z -dimensional tanh output, and the action path is

$$z_t^{b,\text{kp}} = E_{\text{kp}}(\tilde{g}_t^{b,\text{kp}}), \quad a_t = D(z_t^{b,\text{kp}}, \mathcal{N}_o^b(o_{t-K+1:t}^b), m_b). \quad (10)$$

The decoder D , including its FiLM layers, is the morphology-conditioned decoder of the motion interface, and target coordinates remain in the physical scale of the robot. Fig. 3 shows



Figure 5. **MorphMimic on a real Unitree G1.** (a) Dance, (b) walking, and (c) teleoperation, in which the robot follows a human operator. Frames run from left to right at intervals of 1.0 s in (a), 0.5 s in (b), and 2.0 s in (c).

Table I

THE FOUR 40,000-ITERATION SHARING CONFIGURATIONS ON THE 128,786-CLIP COMMON SET. *Specialist* (S) HAS A SEPARATE WORLD MODEL AND POLICY PER ROBOT, *shared WM* (W) SHARES ONLY THE WORLD MODEL, *shared policy* (P) SHARES THE POLICY AND CRITICS, AND *unified* (U) SHARES THE WORLD MODEL, POLICY, AND CRITICS. COMPLETION IS THE FRACTION OF CLIPS THAT REACH THE FINAL FRAME WITHOUT EARLY TERMINATION. HIGHER IS BETTER ($\uparrow$); THE BEST VALUE IN EACH ROBOT-INTERFACE BLOCK IS IN BOLD.

Interface	Configuration	G1		X2 Ultra		T2	
		Completion (%) $\uparrow$	Mean return $\uparrow$	Completion (%) $\uparrow$	Mean return $\uparrow$	Completion (%) $\uparrow$	Mean return $\uparrow$
Full motion	Specialist	99.797	0.9306	99.818	0.9253	99.860	0.9448
	Shared WM	99.734	0.9274	99.845	0.9269	99.861	0.9401
	Shared policy	99.806	0.9317	99.861	0.9278	99.775	0.9404
	Unified	99.747	0.9358	99.891	0.9304	99.886	0.9398
Five-point	Specialist	99.849	0.8807	99.841	0.8689	99.857	0.9014
	Shared WM	99.720	0.8681	99.851	0.8788	99.875	0.8979
	Shared policy	99.769	0.8781	99.847	0.8855	99.766	0.8976
	Unified	99.729	0.8626	99.880	0.8774	99.852	0.8890

walking and waving through this interface on all three robots.

1) *Keypoint Encoder Training*: After the motion-policy update and its critic fit, an additional step trains the keypoint encoder on N_{kp} imagined starts. Let $\mathcal{I}_b$ contain the valid rewarded rollout steps, let $z_{i,t}^{b,motion}$ be the latent that the motion encoder produces from the clean full goal at the corresponding imagined state, and let β_{kp} weight the latent-matching term. The keypoint loss is

$$\mathcal{L}_{kp}^b = -\hat{J}_b^{kp} + \beta_{kp} \mathcal{A}_b, \quad \mathcal{A}_b = \frac{1}{|\mathcal{I}_b|} \sum_{(i,t) \in \mathcal{I}_b} \left\| z_{i,t}^{b,kp} - \text{sg}(z_{i,t}^{b,motion}) \right\|_2^2. \quad (11)$$

The keypoint return $\hat{J}_b^{kp}$ uses the imagined-return objective with the keypoint encoder and a separate critic, computed with the full motion-tracking reward and the clean reference. The alignment term sums over the latent dimensions and averages uniformly over the valid rollout steps. As in SuperMimic's

downstream-encoder update [2, Appendix B], only the keypoint encoder is optimized in this step, and derivatives pass through the fixed decoder, model, and critic. The keypoint critic is then fit on these rollouts. It receives state features, the clean full-motion goal, the reference anchor positions relative to the robot, the context, and the morphology descriptor, while the actor receives the five-point targets.

To expose the encoder to imprecise targets, we perturb the keypoint targets of the actor during imagination with bounded position and orientation jitter, sampled once for the target-frame sequence of a rollout so that overlapping windows reuse the same perturbation for a given frame. Rewards, termination, and critic targets use the clean reference, as do collection and evaluation. A fraction f_{kp} of the collection environments acts through the keypoint encoder and the rest through the motion encoder, and both feed the replay buffer, while the motion update keeps its original batch size and loss. The keypoint update therefore also changes the collected data and hence

the data used to train the world model.

G. Training Across Robots

Each robot runs in its own process with its own simulator, motion library, replay buffer, and reset sampler. For any shared parameter block ω , the optimizer uses

$$\overline{\nabla_{\omega} \mathcal{L}} = \frac{1}{|\mathcal{B}|} \sum_{b \in \mathcal{B}} \nabla_{\omega} \mathcal{L}_b, \quad (12)$$

averaged before SuperMimic’s gradient clipping and Adam updates [2, Table A.5]. State, observation, and goal normalizers, residual scales, body-height statistics, joint limits, default poses, and morphology vectors remain local to each robot and are excluded from parameter synchronization. A checkpoint therefore contains the shared weights together with the statistics and constants for its selected robot. Here, a shared policy means shared encoder and decoder weights combined with these robot-specific buffers and the native action map. In MorphMimic, both encoders, the decoder, and both interface-specific critics share weights across robots, while keypoint normalizers remain robot-specific. Cross-robot alignment in (9) acts on motion latents, and keypoint latents align to the motion latents of their own robot through $\mathcal{A}_b$. After warm-up, each training iteration collects simulator transitions into robot-specific replay buffers, fits the shared world model, updates the motion encoder and decoder, and fits the motion critic. It then updates the keypoint encoder and fits the keypoint critic on a fresh set of imagined starts.

IV. EXPERIMENTS

A. Experimental Setup

We evaluate four sharing configurations: a specialist with separate world model and policy per robot, a shared-world-model configuration, a shared-policy configuration, and a unified configuration that shares the world model, policy, and critics. All configurations use the G1, X2 Ultra, and T2, position-free full-motion and five-point interfaces, 4096 environments per robot, and 40,000 iterations. Architectures and optimizer settings are matched, actuator delay is randomized over 0–40 ms, and critic targets are clipped at the maximum discounted return. The unified run processes three robot streams at once, whereas each specialist processes one.

Evaluation uses the 128,786 clips common to all three training libraries. Each clip runs once from its first frame under the deterministic policy, with nominal simulator parameters and clean observations, resets, and targets. Five-point commands include all five targets and ten preview frames. Completion is the fraction of clips that reach their final frame without early termination. For a clip of L_i frames and $n_i \leq L_i - 1$ executed transitions, the normalized return is

$$R_i^{\text{norm}} = \frac{\sum_{t=0}^{n_i-1} r_{i,t+1}}{r_{\max}(L_i - 1)}. \quad (13)$$

Because the denominator covers the complete clip, early termination lowers the return, and perfect tracking without penalties gives a return of one. We report the mean over clips.

B. Full-Corpus Tracking

Table I reports all four configurations on the common set. Every shared configuration stays within 0.02 mean return of its specialist on all six robot–interface pairs. The largest difference is 0.018 on the G1 five-point interface for the unified configuration. The best completion rate varies by robot and interface: the unified configuration is best on both X2 interfaces and T2 full-motion tracking, while the specialist is best on both G1 interfaces and T2 five-point tracking. Sharing therefore changes which clips terminate without producing a uniform advantage in completion. The unified run takes 5 h 26 min and 16.3 GPU-hours on three H100 NVL GPUs; the three specialists use 13.9 GPU-hours in total.

C. Command Interfaces

Figs. 4 and 3 show full-motion and five-point tracking with the unified policy. Across the four configurations, the five-point interface lowers mean return by 0.04–0.07 on each robot. Its effect on completion depends on the configuration: the specialist loses 67 clips on G1, 29 on X2, and gains 4 on T2, while the unified policy loses 23, 13, and 44 clips, respectively. The smaller target set therefore preserves the command interface while reducing tracking precision.

D. Hardware Experiments

We deploy the simulation-trained MorphMimic policy directly on a physical Unitree G1. The deployed weights are those of the 40,000-iteration MorphMimic policy in Table I, exported to ONNX and identical for the three robots. At each control step, the policy reads the orientation and angular velocity of the pelvis IMU and the 29 joint encoders and sends joint position targets to the PD controllers with the training gains. In Fig. 5(a) and (b), the G1 tracks a dance clip and a walking clip from the reference library through the full-motion interface. Each clip is rotated to the heading of the robot at its start, so that the IMU heading aligns the reference with the robot. In Fig. 5(c), an operator wearing a PICO headset, two hand controllers, and two ankle trackers teleoperates the robot through the five-point interface. The operator’s motion is resampled and retargeted to G1 reference states at the control rate, and a nominal buffer of $K - 1$ frames supplies the preview window from the received reference states. The G1 follows the operator through standing, squatting, and arm motions in real time. These experiments demonstrate that the position-free command representation and the shared policy transfer from simulation to hardware, with policy feedback from the onboard joint encoders and IMU.

V. CONCLUSION

MorphMimic trains one differentiable world model and one whole-body tracking policy for the Unitree G1, AgiBot X2 Ultra, and Booster T2. A canonical joint and body representation maps the robots to a common coordinate system, a physical descriptor conditions the shared networks on actuation, geometry, and mass, and latent command alignment relates corresponding motions across robots. A five-point interface

for pelvis, ankle, and wrist targets learns to reproduce the full-motion latents and uses the same decoder. On 128,786 reference clips common to the training libraries, all three shared configurations remain within 0.02 mean return of the single-robot specialists on both interfaces. Completion rates vary by robot and sharing choice, and the five-point interface lowers mean return in every configuration.

Because both interfaces express target positions in the reference frame, the policy uses joint-encoder and IMU-based orientation feedback. The shared policy therefore dances, walks, and follows a teleoperator on a physical G1 using only onboard joint encoders and the IMU. These experiments show that embodiment information can be provided to one world model and one policy. Additional humanoids can be added through the common representation, and additional command interfaces can use the aligned latent space. Future work will extend the hardware experiments to the X2 Ultra and T2 and the representation and descriptor to further humanoids.

REFERENCES

- [1] Z. Luo, Y. Yuan, T. Wang, C. Li, F. Castañeda, S. Chen, Z.-A. Cao, J. Li, D. Minor, Q. Ben, J. Park, D. Sami, Z. Wang, X. Da, R. Ding, C. Hogg, L. Song, E. Lim, E. Jeong, T. He, H. Xue, W. Xiao, S. Yuen, J. Kautz, Y. Chang, U. Iqbal, L. J. Fan, and Y. Zhu, “SONIC: Supersizing motion tracking for natural humanoid whole-body control,” *Science Robotics*, vol. 11, no. 117, p. ead4592, 2026. [Online]. Available: <https://doi.org/10.1126/scirobotics.aed4592>
- [2] Q. Liao, Y. Shao, J. Long, K. Zakka, R. Horowitz, P. Abbeel, and K. Sreenath, “SuperMimic: Scaling humanoid whole-body control with one differentiable world model,” in *Conference on Robot Learning (CoRL)*, 2026, accepted to CoRL 2026 (oral). [Online]. Available: <https://arxiv.org/abs/2610.28480>
- [3] M. Yang, T. Yu, F. Li, and H. Chen, “Any2Any: Efficient cross-embodiment transfer for humanoid whole-body tracking,” *arXiv preprint arXiv:2605.23733*, 2026. [Online]. Available: <https://arxiv.org/abs/2605.23733v3>
- [4] T. Chen, A. Murali, and A. Gupta, “Hardware conditioned policies for multi-robot transfer learning,” in *Advances in Neural Information Processing Systems*, vol. 31, 2018. [Online]. Available: https://papers.neurips.cc/paper_files/paper/2018/hash/b8c9bf77a3d250a4523ba67a65a7d031-Abstract.html
- [5] N. Bohlinger, G. Czechmanowski, M. P. Krupka, P. Kicki, K. Walas, J. Peters, and D. Tateo, “One policy to run them all: an end-to-end learning approach to multi-embodiment locomotion,” in *Proceedings of the 8th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, vol. 270, 2025, pp. 3356–3378. [Online]. Available: <https://proceedings.mlr.press/v270/bohlinger25a.html>
- [6] N. Bohlinger and J. Peters, “Multi-embodiment locomotion at scale with extreme embodiment randomization,” *arXiv preprint arXiv:2509.02815*, 2025. [Online]. Available: <https://arxiv.org/abs/2509.02815>
- [7] Y. Xue, Y. Lin, W. Dong, Y. Tang, J. Wang, J. Pang, M. Zhou, M. Liu, and W. Zhang, “Scalable and general whole-body control for cross-humanoid locomotion,” *arXiv preprint arXiv:2602.05791*, 2026. [Online]. Available: <https://arxiv.org/abs/2602.05791v3>
- [8] M. H. Danesh, C. Li, A. Abyaneh, A. Houssaini, K. Ellis, G. Berseth, M. Hutter, and H.-C. Lin, “Morphology-conditioned world model for cross-embodiment quadrupedal locomotion,” *arXiv preprint arXiv:2604.08780*, 2026. [Online]. Available: <https://arxiv.org/abs/2604.08780v2>
- [9] Y. Wang, J. Kong, S. Wei, X. Li, H. Lin, H. Zhao, T. Zhou, L. Gan, and H. Shao, “WestWorld: A knowledge-encoded scalable trajectory world model for diverse robotic systems,” in *International Conference on Machine Learning (ICML)*, 2026.
- [10] J. Zhang, C. Gu, and L. Zhang, “X-WBC: A cross-embodiment foundation model for humanoid whole-body control,” in *Conference on Robot Learning (CoRL)*, 2026, accepted to CoRL 2026. [Online]. Available: <https://arxiv.org/abs/2609.15213>
- [11] W. Huang, I. Mordatch, and D. Pathak, “One policy to control them all: Shared modular policies for agent-agnostic control,” in *Proceedings of the 37th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 119, 2020, pp. 4455–4464. [Online]. Available: <https://proceedings.mlr.press/v119/huang20d.html>
- [12] A. Gupta, L. Fan, S. Ganguli, and L. Fei-Fei, “MetaMorph: Learning universal controllers with transformers,” in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=OpmqtK_GvYL
- [13] G. Feng, H. Zhang, Z. Li, X. B. Peng, B. Basireddy, L. Yue, Z. Song, L. Yang, Y. Liu, K. Sreenath, and S. Levine, “GenLoco: Generalized locomotion controllers for quadrupedal robots,” in *Proceedings of the 6th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, vol. 205, 2023, pp. 1893–1903. [Online]. Available: <https://proceedings.mlr.press/v205/feng23a.html>
- [14] S. Wei, X. Chen, F. Xie, G. E. Katz, Z. Gan, and L. Gan, “Beyond topology: A morphological symmetry graph representation for locomotion policy learning,” *arXiv preprint arXiv:2512.00727*, 2025. [Online]. Available: <https://arxiv.org/abs/2512.00727>
- [15] I. Georgiev, V. Giridhar, N. Hansen, and A. Garg, “PWM: Policy learning with multi-task world models,” *arXiv preprint arXiv:2407.02466*, 2024, version 3, revised February 2025. [Online]. Available: <https://arxiv.org/abs/2407.02466v3>
- [16] F. Xie, S. Wei, Y. Song, Y. Yue, and L. Gan, “Morphological-symmetry-equivariant heterogeneous graph neural network for robotic dynamics learning,” in *Proceedings of the 7th Annual Learning for Dynamics & Control Conference*, ser. Proceedings of Machine Learning Research, vol. 283, 2025, pp. 1392–1405. [Online]. Available: <https://proceedings.mlr.press/v283/xie25a.html>
- [17] Bones Studio, “BONES-SEED: Skeletal everyday embodiment dataset,” Hugging Face dataset, accessed October 22, 2026. [Online]. Available: <https://huggingface.co/datasets/bones-studio/seed>
- [18] NVIDIA, “SOMA Retargeter,” Software repository, experiments use commit cdbb2b4; accessed October 22, 2026. [Online]. Available: <https://github.com/NVIDIA/soma-retargeter>
- [19] E. Perez, F. Strub, H. de Vries, V. Dumoulin, and A. Courville, “FiLM: Visual reasoning with a general conditioning layer,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018. [Online]. Available: <https://ojs.aaai.org/index.php/AAAI/article/view/11671>

APPENDIX A IMPLEMENTATION DETAILS

Table II

ROBOT MODELS. SEGMENT LENGTHS ARE DISTANCES BETWEEN BODY ORIGINS AT THE MODEL’S ZERO POSE.

	G1	X2 Ultra	T2
Body joints	29	29	29
Head joints outside policy	0	2	2
Total mass (kg)	33.3	43.5	42.0
Thigh length (m)	0.341	0.327	0.347
Shank length (m)	0.300	0.282	0.336
Upper arm length (m)	0.185	0.205	0.218
Forearm length (m)	0.184	0.203	0.212

Table III
IMPLEMENTATION PARAMETERS

Symbol	Description	Value
<i>Task and representation</i>		
Δt	Control interval	0.02 s
γ	Discount factor	0.99
$r_{\max}$	Maximum reward per step	0.1
$\bar{e}_z, \bar{e}_u$	Termination thresholds	0.25 m, 0.8
N_j, N_b	Body joints, tracked bodies	29, 14
K	History and goal frames	10
κ_a	Action scale	0.25
ρ	Soft joint-limit fraction	0.9
d_m, d_ξ	Descriptor and context sizes	125, 33
<i>World model</i>		
H_m	Model-fitting horizon	8 steps
S_ξ	Friction, bias, and CoM scaling	1, 100, 20
k_z^0, k_u^0	Initial hazard gains	100 m ⁻¹ , 20
c_k	Hazard-gain cap	10
<i>Policy and alignment</i>		
d_z	Latent command size	32
H_π	Imagination horizon	32 steps
λ	λ -return parameter	0.95
σ_p	Position-reward kernel width	0.3 m
N_a, β_a	Alignment pairs and weight	1024, 0.05
<i>Five-point interface</i>		
–	Encoder hidden widths	2048, 1024, 512, 256
N_{kp}	Imagined starts per update	4096
f_{kp}	Keypoint share of environments	0.5
β_{kp}	Latent-matching weight	0.1
–	Position jitter (horiz., vert.)	± 0.05 m, ± 0.01 m
–	Rotation jitter (roll–pitch, yaw)	± 0.1 rad, ± 0.2 rad

Table III lists the constants of Section III. The proprioceptive observation has 93 values per frame and the full-motion goal 67 values per frame, so the policy receives 930 history values and 670 goal values over the K -frame window. The world-frame state has 240 values, or 268 features with the two-column rotation representation, and the dynamics network receives 472 inputs. The critics receive $3K = 30$ reference-relative anchor coordinates, and the five-point encoder receives 465 inputs, namely 450 target features and 15 mask bits. The policy and critic networks follow SuperMimic’s Table A.4 with layer normalization in the dynamics and critics. The

Adam updates use the single-robot learning rate, with one GPU per robot process. In the 40,000-iteration comparisons, critic targets are clamped at the maximum discounted return, $G_t \leftarrow \min(G_t, r_{\max}/(1 - \gamma))$ with $r_{\max}/(1 - \gamma) = 10$; critic outputs remain unconstrained so that the actor receives gradients through the value function. The residual scales $\sigma_{\Delta, b}$ have positive per-feature floors, the inverse state transform orthonormalizes the predicted rotation columns, and the hazard gains start at k_z^0 and k_u^0 , are capped at c_k times these values, and start with zero biases. The keypoint jitter ranges follow Table S4 of SONIC [1].

The canonical map of (2) is checked with forward kinematics for the direction of matched body motion and the alignment of neutral poses. On T2, the joint named `elbow_pitch` supplies humerus twist and `elbow_yaw` supplies elbow flexion, and the shoulder neutral pose requires a 90° offset. The X2 waist uses a different Euler order, which the map aligns slot by slot. For T2, the pelvis equivalent is the waist yaw link while the floating base is the trunk, and the IMU body is the pelvis on G1 and X2 and the trunk on T2. Reference-state initialization uses the native floating-base pose, and each robot applies its configured actuator model and effort limits. X2 and T2 have two additional head joints, which remain at their default pose, and the masses and lengths in Table II are measured on the compiled simulation models. For the five-point interface, ankle targets use the ankle-roll links, wrist targets use the distal link of each wrist chain, which is the wrist-yaw link on G1 and the wrist-roll link on X2 and T2, and the T2 pelvis target uses the waist yaw link. The pelvis rotation comes from the IMU attitude estimate on G1 and X2, and on T2 it is reconstructed from the trunk IMU attitude and the waist joint encoders.

The SOMA Retargeter matches human joints to robot links using weighted position and orientation objectives with a calibrated scale and offsets, a foot-contact post-process adjusts planted feet, and robot-specific configurations account for the X2 and T2 kinematics. Its 120 Hz output is converted to 50 Hz reference states with the forward kinematics of the robot and feasibility filters. Inverse kinematics uses the soft joint limits

$$q_j^{\text{lo,soft}} = q_j^{\text{mid}} - \rho \bar{q}_j, \quad q_j^{\text{hi,soft}} = q_j^{\text{mid}} + \rho \bar{q}_j, \quad (14)$$

where q_j^{mid} and $\bar{q}_j$ are the midpoint and half-range of the joint limits $[q_j^{\text{lo}}, q_j^{\text{hi}}]$, which keep the references clear of clipping at reset and of the joint-limit penalty of the controller. With these limits, the fraction of left-knee frames outside them on the checked subsets falls from 30% for X2 and 38% for T2 to zero, and all checked joints of both robots lie within the limits. From 129,398 source clips, conversion and feasibility filtering produce 128,981 X2 clips and 129,048 T2 clips, and the G1 library contains the 129,398 provided retargets. On the physical G1, the stand that precedes a clip pitches both ankles back by 0.067 rad, which places the torso over the feet, and the teleoperation buffer of $K - 1$ frames corresponds to 180 ms.