

281 Appendix

282 A Expert Training

283 A.1 Observation and Action Space

284 The observation space of the expert policies includes the joint angles, joint velocities, previous ac-
285 tions, trunk angular velocities, gravity vector and the command velocities. The observation space of
286 the critics includes the same observations as for the policies, but also includes privileged informa-
287 tion: the trunk linear velocity, trunk height over the ground, feet contact states and feet air times.

288 The policies control the robots at 50 Hz with a PD controller, where the target joint angles are
289 generated by scaling the action of the policy and adding it to the nominal joint configuration of the
290 robot: $q_{\text{target}} = q_{\text{nominal}} + \sigma \cdot a$. We define the nominal joint configuration as a standing pose of a robot
291 and use the same configuration for all robots of the same morphology class (see Appendix B.3). For
292 the action scaling factor σ , we use 0.3 for quadrupeds and hexapods, and 0.75 for humanoids. For
293 the PD controller, we use $K_p = 20$ and $K_d = 0.5$ for quadrupeds, $K_p = 25$ and $K_d = 0.5$ for
294 hexapods, and $K_p = 60$ and $K_d = 2.0$ for humanoids.

295 A.2 Domain Randomization

296 To enable sim-to-real transfer of the trained policies, we add strong domain randomization during
297 training. We use a performance-based curriculum learning approach, where the domain randomiza-
298 tion ranges are increased from 0 (or their mean if not zero-centered) to the final values in Table 1 over
299 the course of training. This curriculum approach allows the policy to learn basic locomotion first
300 in the simplest possible environment before adapting to wider variations. We define a curriculum
301 coefficient from 0 to 1, which is multiplied with the domain randomization ranges (and the reward
302 penalty coefficients). The coefficient of an environment is increased by 0.01 if the policy completed
303 the episode without falling, and the average tracking error of the target x,y velocity is below 0.4 m/s,
304 and the coefficient is reduced by 0.01 otherwise.

305 Every embodiment in GENBOT-1K uses the same domain randomization ranges. The "starting"
306 values (naming scheme in Table 1) are sampled uniformly at the start of every episode to randomize
307 the starting state of the robot. The "noise" values are sampled uniformly for every simulation step
308 to add noise to the observations. The values of every other parameter are sampled uniformly every
309 simulation step with a probability of 0.002 (on average every 500 steps / every 10 seconds). Pushes
310 are applied as linear velocities to the trunk of the robot.

Table 1: **Domain randomization configuration.** Domain randomization values and ranges for every randomized parameter during the expert RL training. The values in the table are the maximum values and ranges in the curriculum when reaching the final curriculum coefficient of 1.

Parameter	Value
Max action delay	1
Chance for action delay	0.05
Min & max motor strength	(0.5, 1.5)
Min & max P gain factor	(0.5, 1.5)
Min & max D gain factor	(0.5, 1.5)
Min & max joint position offset	(-0.05, 0.05)
Min & max starting orientation factor	(-0.0625, 0.0625)
Min & max starting joint position factor	(-0.5, 0.5)
Min & max starting joint velocity factor	(-0.5, 0.5)
Min & max starting linear velocity	(-0.5, 0.5)
Min & max starting angular velocity	(-0.5, 0.5)
Joint position noise	0.01
Joint velocity noise	1.5
Angular velocity noise	0.2
Gravity velocity noise	0.05
Joint observation dropout chance	0.05
Min & max static friction	(0.05, 2.0)
Min & max dynamic friction	(0.05, 1.5)
Min & max restitution	(0.0, 1.0)
Min & max added mass	(-2.0, 2.0)
Min & max gravity	(-8.81, 10.81)
Min & max joint friction	(0.0, 0.01)
Min & max joint armature	(0.0, 0.01)
Min & max pushes in x	(-1.0, 1.0)
Min & max pushes in y	(-1.0, 1.0)
Min & max pushes in z	(-1.0, 1.0)

A.3 Reward Function

Table 2 contains all reward terms and coefficients of the reward function for the expert training of all robots in the GENBOT-1K dataset. Joint-based (T6-T12) and feet-based (T14-T17) reward terms are calculated as the mean over every joint and foot, respectively, to account for the varying amounts of joints and feet of the generated embodiments. The coefficients of all penalties are attached to the curriculum coefficient (see Appendix A.2) and thus linearly increase from 0 to the final values in Table 2 over the course of training. This makes the training process less sensitive to the precise values of the coefficients.

Table 2: **Reward terms for the RL training of embodiment-specific experts.** All reward terms and the corresponding coefficients that compose the reward function for the expert training. While all the coefficients work for all embodiments, for the final experiments, we tweaked four coefficients for the humanoid embodiments to improve the style of the gait: ^{*1} 3.0, ^{*2} 1.5, ^{*3} 43.2, ^{*4} 6e-3.

	Term	Equation	Coefficient
T1	Xy velocity tracking	$\exp(- v_{xy} - c_{xy} ^2/0.25)$	2.0 ^{*1}
T2	Yaw velocity tracking	$\exp(- \omega_{yaw} - c_{yaw} ^2/0.25)$	1.0 ^{*2}
T3	Z velocity penalty	$- v_z ^2$	2.0
T4	Pitch-roll velocity penalty	$- \omega_{pitch, roll} ^2$	0.05
T5	Pitch-roll position penalty	$- \theta_{pitch, roll} ^2$	5.0
T6	Joint nominal differences penalty	$- q - q^{\text{nominal}} ^2$	14.4 ^{*3}
T7	Joint position limits penalty	$-\mathbb{1}(0.9q_{\min} < q < 0.9q_{\max})$	120.0
T8	Joint velocity limits penalty	$-\mathbb{1}(0.9\dot{q}_{\min} < \dot{q} < 0.9\dot{q}_{\max})$	10.0
T9	Joint accelerations penalty	$- \ddot{q} ^2$	5e-6
T10	Joint torques penalty	$- \tau ^2$	2.4e-4
T11	Action rate penalty	$- a_t - a_{t-1} ^2$	0.12
T12	Action smoothness penalty	$- a_t - 2a_{t-1} + a_{t-2} ^2$	0.12
T13	Walking height penalty	$- h - h_{\text{nominal}} ^2$	30.0
T14	Air time penalty	$-\sum_f \mathbb{1}(p_f)(p_f^T - 0.5)$	0.1
T15	Symmetry penalty	$-\sum_f \mathbb{1}(p_f^{\text{left}})\mathbb{1}(p_f^{\text{right}})$	0.5
T16	Feet y distance penalty	$- f_{y \text{ distance}}^{\text{actual}} - f_{y \text{ distance}}^{\text{target}} ^2$	2.0
T17	Feet force penalty	$- f_{\text{force}} ^2$	8e-3 ^{*4}
T18	Self-collision penalty	$-\mathbb{1}_{\text{self-collision}}$	1.0

319 A.4 PPO Hyperparameters

320 We use the same PPO hyperparameters for the training of all expert policies, detailed in Table 3.
 321 Searching for better hyperparameters for every embodiment might lead to increased performance
 322 but is impractical when considering training ~ 1000 embodiments. The chosen hyperparameters are
 323 based on common practices in legged locomotion research [39, 72] and preliminary tuning on a
 324 small subset of embodiments.

Table 3: **PPO hyperparameters for expert policy training.**

Hyperparameter	Value
Batch size	98304
Mini-batch size	24576
# epochs	5
Initial learning rate	0.001
Learning rate schedule	Adaptive with target KL 0.01
Entropy coefficient	0.002
Discount factor	0.99
GAE λ	0.95
Clip range	0.2
Max gradient norm	1.0
Initial action standard deviation	1.0
Clip range action mean	-10.0, 10.0
Policy and critic hidden layers	[512, 256, 128]
Activation function	ELU
# training iterations	17500 (quadruped, hexapod), 42500 (humanoid)

B Embodiment Generation

B.1 Basic Units for Quadruped, Humanoid and Hexapod

Tables 4 and 5 provide the base values for geometry-related and kinematics-related parameters, respectively, for representative links across quadruped, humanoid, and hexapod morphologies. For the humanoid class, we report parameters for the trunk and the left-side lower-body links. For quadruped and hexapod classes, we include the front-left leg. The remaining components are either symmetric or peripheral to locomotion (e.g., arms or head for humanoids) and therefore omitted.

The base parameter values are partially inspired by the Unitree Go2 and H1 platforms, offering a degree of realism without exact replication. This design choice is consistent with prior work such as GenLoco [55], which abstracts physical characteristics from real robots to define a diverse yet grounded design space. Robots instantiated with these values correspond to a $1.0\times$ variation setting (i.e., no geometric, kinematic, or topological scaling applied), and serve as the reference point for applying the variation factors listed in Tables 4 and 5.

To support meaningful evaluation of generalization, these reference robots are excluded from the training set. Every robot in the training set differs from Go2 and H1 by at least one geometric, topological, or kinematic variation, along with additional discrepancies due to loose alignment in parameter values (e.g., each joint in the humanoid closest to H1 differs by a few centimeters, and the overall height differs by approximately 10 cm). This diversity encourages the learned policy to capture broadly transferable motion patterns. As discussed in Section 4, empirical results suggest that the policy has acquired sufficiently generalizable behaviors to support both cross-embodiment and sim-to-real transfer, which is generally considered highly challenging.

B.2 Generation Algorithm

We construct each robot embodiment in a tree-like structure by iteratively connecting links using joints, following the URDF specification and the basic units described in Section B.1. The construction procedure varies slightly across morphologies:

- **Humanoids:** The root node is the pelvis. We first append the torso and hip links, then attach the shoulder and arm links for the upper body, followed by the thigh, calf, and foot links for the lower body.
- **Quadrupeds and hexapods:** The root node is the trunk. We sequentially append the hip links to the trunk, then connect the leg and foot links to form the complete body.

To ensure diversity in the generated embodiments, we introduce variations in geometry, topology, and kinematics during the construction process, as detailed in Section 3. Table 6 summarizes the variation parameters and their corresponding candidate values. While most parameters are self-explanatory, we clarify a few specific cases:

- **Number of knee joints:** If a leg is configured with zero knee joints, the calf link is omitted, and the thigh link is directly connected to the foot.
- **Foot link size:** For humanoids, foot links are modeled as boxes and scaled by length; for quadrupeds and hexapods, foot links are modeled as spheres and scaled by radius.
- **Joint limit variation:** Joint limits are varied by uniformly scaling the nominal joint ranges about the nominal joint position, which serves as a fixed point.

B.3 Nominal Joint Configurations

Nominal joint configurations are used to initialize robot poses during training, contribute to reward terms that discourage deviations too far from these default joint angles, and function as offsets to the actions of the expert and distillation policies. As such, they serve as useful regularizers for learning realistic and efficient gaits. To support scalability across diverse morphologies, we generate nominal

370 configurations by reusing unit values across the generated embodiments. The nominal joint angles
371 used are summarized in Table 7.

Table 4: **Base geometry and mass parameters for representative link types in the embodiment generation pipeline used in GENBOT-1K.** Geometry dimensions are specified according to shape type: Sphere (radius), Cylinder (length, radius), and Box (length, width, height).

Class	Link Name	Geometry Type	Geometry Dimension (m)	Mass (kg)
Humanoid	Pelvis	Sphere	(0.05,)	5.390
	Torso	Box	(0.08, 0.26, 0.18)	17.789
	Hip yaw link	Cylinder	(0.02, 0.01)	2.244
	Hip roll link	Cylinder	(0.01, 0.02)	2.232
	Thigh	Cylinder	(0.2, 0.05)	4.152
	Calf	Cylinder	(0.2, 0.05)	1.721
	Foot	Box	(0.28, 0.03, 0.024)	0.474
Quadruped	Trunk	Box	(0.38, 0.09, 0.11)	6.921
	Hip	Cylinder	(0.04, 0.046)	1.152
	Thigh	Box	(0.21, 0.025, 0.034)	1.152
	Calf	Cylinder	(0.12, 0.013)	0.154
	Foot	Sphere	(0.022,)	0.040
g	Trunk	Box	(0.8, 0.5, 0.1)	6.921
	Hip	Sphere	(0.05,)	0.678
	Thigh	Cylinder	(0.22, 0.03)	1.152
	Calf	Cylinder	(0.22, 0.025)	0.154
	Foot	Sphere	(0.03,)	0.040

Table 5: **Motor and joint properties of the generated embodiments in GENBOT-1K.**

Class	Joint Name	Joint Limits (rad)	Max. Torque (N·m)	Max. Velocity (rad/s)
Humanoid	Torso joint	(-2.35, 2.35)	200	23
	Shoulder pitch joint	(-2.87, 2.87)	40	9
	Shoulder roll joint	(-0.34, 3.11)	40	9
	Shoulder yaw joint	(-1.30, 4.45)	18	20
	Elbow joint	(-1.25, 2.61)	18	20
	Hip yaw/roll joint	(-0.43, 0.43)	200	23
	Hip pitch	(-3.10, 2.50)	200	23
	Knee joint	(-0.26, 2.00)	300	14
	Ankle joint	(-0.87, 0.52)	40	9
Quadruped	Hip pitch joint	(-1.05, 1.05)	23.7	30.1
	Front thigh joint	(-1.57, 3.49)	23.7	30.1
	Rear thigh joint	(-0.52, 4.53)	23.7	30.1
	Knee joint	(-2.72, -0.84)	45.43	15.7
Hexapod	Hip joint	(-1.57, 1.57)	100	30
	Thigh joint	(-1.57, 1.57)	100	30
	Knee joint	(-1.57, 1.57)	100	30

Table 6: **Variation parameters across geometry, topology, and kinematics in the embodiment generation algorithm.** The torso link randomization only applicable to the humanoid class.

Variation Type	Parameter Name	Candidate Values
Topology	Number of knee joints	{0, 1, 2, 3}
Geometry	Scaling factor for all link size	{0.8, 1.0, 1.2}
	Scaling factor for thigh link length	{0.4, 0.8, 1.0, 1.2, 1.6}
	Scaling factor for calf link length	{0.4, 0.8, 1.0, 1.2, 1.6}
	Scaling factor for foot link size	{1.0, 2.0}
	Scaling factor for torso link size	{0.4, 0.8, 1.0, 1.2, 1.6}
Kinematics	Scaling factor for knee joint limits	{0.2, 0.6, 1.0}

Table 7: **Nominal joint configurations for generated embodiments in GENBOT-1K.** These joint angles are used to initialize robot poses, define regularization rewards, and function as offsets to the policy actions. The values are consistent across symmetric limbs.

Class	Joint Name	Joint Angle (rad)
Humanoid	Torso	0.0
	Shoulder (Left/Right, pitch/roll/yaw)	0.0
	Elbow (Left/Right)	0.0
	Hip pitch (Left/Right)	-0.4
	Hip roll/yaw (Left/Right)	0.0
	Knee (Left/Right)	0.8
	Ankle (Left/Right)	-0.4
Quadruped	Hip (Front/Rear, Left/Right)	± 0.1
	Thigh (Front, Left/Right)	0.8
	Thigh (Rear, Left/Right)	1.0
	Knee (Front/Rear, Left/Right)	-1.5
	Additional knee joints (if any)	0.0
Hexapod	Hip (Front/Middle/Rear, Left/Right)	0.0
	Thigh (Front/Middle/Rear, Left/Right)	0.79
	Knee (Front/Middle/Rear, Left/Right)	0.79
	Additional knee joints (if any)	0.0

C Cross-Embodiment Distillation

C.1 Expert Data Collection

For every embodiment, we run the expert RL policy for 600 simulation steps using 4096 parallel environments. This results in a total of 1,985,740,800 data samples across all training embodiments. Note that the episode length during the expert training is 1000 simulation steps (equivalent to 20 physical seconds), thus, the collected data only covers the first half of the episode. Using the full length may provide more time-correlated data, which we did not analyze due to time constraints. The final dataset needs around 5 TB of storage using the h5py format without additional compression.

C.2 URMA Architecture Details

The observation space of the URMA policy is split into two parts: joint-specific observations o_j and general observations o_g . The joint-specific observations o_j include the joint angle, joint velocity, previous action of the joint (shape: $(j(e), 3)$). The general observations o_g include the trunk linear velocity, gravity vector, command velocities, PD gains, action scaling factor, total mass of the robot, robot dimensions, number of joints and feet size (shape: $(20,)$).

The description vectors d_j of the joints include the relative cartesian position of the joint in the nominal configuration, joint rotation axis, joint nominal angle, maximum joint torque, maximum joint velocity, joint position limits, p-gain, d-gain and action scaling factor, robot mass and dimensions (shape: $(j(e), 18)$).

We build on the original URMA neural network architecture, as shown in Figure 7, from Bohlinger et al. [39] with the following modifications:

- We use multi-headed attention for the encoding of the joint observations and descriptions to increase the expressiveness of the policy. All our experiments use 3 attention heads.
- We remove the feet-specific attention encoder as not all robots in the real world have foot-specific sensors, like pressure sensors.
- We directly use the output from the action decoder μ_ν as the action of the policy, instead of using an additional head to produce a standard deviation and sampling from a Gaussian distribution, as we train the policy with imitation learning instead of RL.
- We add another encoding layer to the general observations o_g to project them into a higher dimensional latent space before concatenating them with all the joint latent vectors from the attention heads.
- We use wider feedforward layers ($2\times$ the hidden dimensions) throughout the network.

The resulting model has 2.1 million parameters. Overall, it is a compact network with strong inductive biases that leverage the compositional structure of robots.

When applying the actions of the policy to the robots, we use the same PD controllers with the same nominal joint configurations and action scaling factors as in the expert training (see Appendix A.1).

C.3 Train-Test Set Splits

We split GENBOT-1K into a training set (80%) and a test set (20%) using a deterministic pseudo-random sampler with a fixed seed, ensuring full reproducibility. The same sampling procedure is applied independently to each morphology class, except for quadrupeds and hexapods, which share identical splits due to matched dataset sizes. Detailed test indices are listed in Table 8, and summary statistics for each category are shown in Table 9.

Table 8: **Train-test splits of GENBOT-1K.** Each index refers to one unique embodiment in each embodiment class. The training set is simply the complement of the test set and thus omitted.

Class	Test Set
Humanoid	[0, 7, 12, 20, 31, 32, 37, 41, 46, 47, 48, 50, 51, 55, 63, 71, 72, 75, 97, 104, 111, 113, 122, 124, 128, 132, 133, 144, 149, 154, 155, 158, 161, 163, 166, 169, 170, 181, 183, 197, 204, 207, 215, 222, 226, 229, 241, 244, 248, 250, 252, 258, 260, 261, 266, 272, 276, 278, 280, 282, 286, 290, 298, 308, 312, 313, 316, 320, 327, 342]
Quadruped	[0, 7, 8, 20, 31, 32, 37, 41, 46, 47, 48, 50, 51, 55, 71, 72, 75, 97, 104, 111, 113, 122, 124, 128, 132, 133, 144, 149, 154, 155, 158, 161, 163, 166, 169, 170, 181, 183, 197, 204, 207, 215, 222, 226, 229, 241, 244, 248, 250, 252, 258, 260, 261, 266, 272, 278, 280, 282, 286, 290, 298, 308, 312, 313, 316, 320, 327]
Hexapod	[0, 7, 8, 20, 31, 32, 37, 41, 46, 47, 48, 50, 51, 55, 71, 72, 75, 97, 104, 111, 113, 122, 124, 128, 132, 133, 144, 149, 154, 155, 158, 161, 163, 166, 169, 170, 181, 183, 197, 204, 207, 215, 222, 226, 229, 241, 244, 248, 250, 252, 258, 260, 261, 266, 272, 278, 280, 282, 286, 290, 298, 308, 312, 313, 316, 320, 327]

Table 9: **Statistics of train-test splits of GENBOT-1K.** The splits have an approximately balanced distribution over different categories.

Class	Total Number	Train Set (80%)	Test Set (20%)
Humanoid	348	278	70
Quadruped	332	265	67
Hexapod	332	265	67
Total	1012	808	204

413 C.4 Training Details

414 We designed an efficient training pipeline that balances disk I/O, CPU preprocessing, GPU utilization,
 415 and RAM usage. Instead of loading every minibatch directly from disk, we first load a fixed
 416 number of data slices, each containing a small subset of steps from multiple robot embodiments,
 417 into an in-memory buffer. Each slice consists of 100 trajectories with 128 steps per trajectory. Once
 418 the buffer is filled, minibatches are sampled uniformly at random, without replacement, until every
 419 sample has been seen a fixed number of times. This strategy reduces disk access overhead, improves
 420 memory locality, and maintains sample diversity throughout training, though it may introduce local
 421 overfitting and biased gradient estimates.

422 Because data from different robots have varied observation and action spaces, we load them sepa-
 423 rately and use gradient accumulation to reduce bias in the gradient estimation. Specifically, gradients
 424 are accumulated across multiple minibatches before each optimizer step, helping to balance contri-
 425 butions across robot embodiments. While effective, this approach still suffers from local gradient
 426 bias. A more principled solution would involve zero-padding to form large, uniform batches across
 427 robots, but implementing this would require architectural and pipeline-level changes, which we did
 428 not pursue due to time constraints. In theory, this could lead to smoother optimization and potentially
 429 better final performance.

430 To ensure numerical stability, we apply gradient clipping with a maximum norm of 5. We use the
 431 AdamW optimizer [109] with $\beta_1=0.9$, $\beta_2=0.999$, and a cosine-annealed weight decay schedule that
 432 decays from 3×10^{-4} to 0 over the course of training [110]. The key hyperparameters for distillation
 433 are summarized in Table 10.

434 Our pipeline requires 128 GB of RAM to maintain the in-memory buffer. Due to the small size of
 435 the URMA policy, training can be efficiently performed on a single GPU (e.g., NVIDIA RTX 4090
 436 or H100). We did not observe significant gains in convergence from increasing batch size, possibly

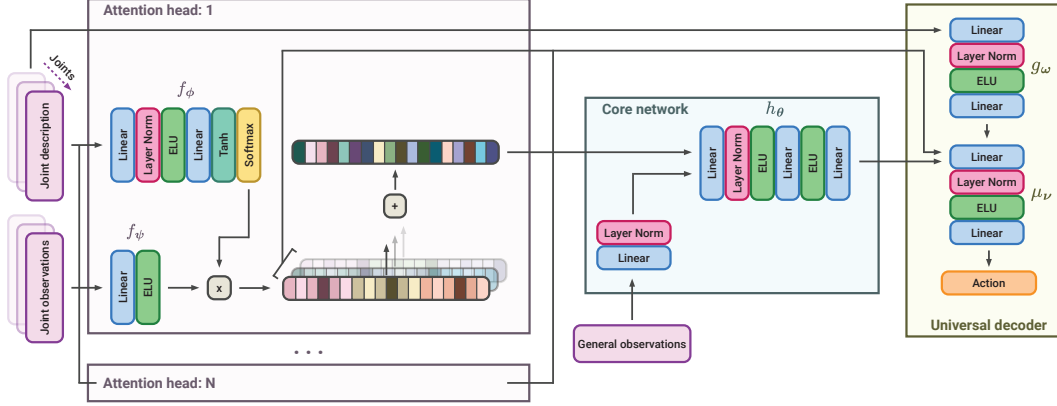


Figure 7: **URMA with multi-head attention.** We extend the original URMA module [39] with multiple attention heads, each aggregating information from joint observations using distinct attention distributions. This design enables the model to capture multi-modal dependencies and improves its capacity to scale across diverse embodiments.

Table 10: **Hyperparameters of the distillation pipeline.**

Hyperparameter	Value
# training samples per embodiment	500×4096
Validation set size	100×4096
Batch size	64
Gradient accumulation steps	8
Gradient clipping threshold	5
Data slice size	100×128
Max slices in buffer	1024
Buffer repeat factor	3
Optimizer	AdamW [109]
AdamW betas	(0.9, 0.999)
Weight decay schedule	$3 \times 10^{-4} \rightarrow 0$ (cosine)
Learning rate schedule	Cosine annealing [110]
# epochs	80

437 due to the structured nature and potential bottlenecks in the model architecture. Further investigation
 438 into the scaling behavior of the training dynamics is left for future work.

439 D Additional Details on Real-World Deployment

440 D.1 Hardware Setup

441 We evaluated our distilled URMA policy zero-shot on two real-world platforms: the Unitree Go2
 442 quadruped and the Unitree H1 humanoid. For each robot, we used its URDF to produce the em-
 443 bodiment description vectors d_j . Before deployment, the policy was converted to the ONNX format
 444 to load it in JAX and guarantee maximum inference speed. The policy inference ran on a Ubuntu
 445 22.04 laptop (Ryzen 9 CPU), interfaced to the robot over a dedicated Ethernet connection. We ran
 446 the control loop at the same 50 Hz and with the same PD gains as in simulation, and sent the target
 447 joint angles to the robot’s internal controller. We limited the commanded x-y-yaw velocity to 0.8 m/s
 448 for the Go2 and 0.5 m/s for the H1, to ensure the robot’s stability and safety during the experiments.

D.2 Implementing Joint Limit Variations

To probe robustness under kinematic constraints, we impose an artificial knee-joint range limited to 20 % of its nominal span. In simulation, one can enforce such limits by directly clamping joint angles within the physics engine; in hardware, however, neither the robot’s encoders nor its embedded PD controller can be modified. Consequently, we introduce a software-level joint-limit layer into the control loop in order to restrict the target joint angle for affected knee joints to the new limits. At each control step, the policy’s commanded knee angle is constrained to the prescribed ± 20 % bounds. Instead, we implemented a software-based solution that restricts the target joint angle for affected knee joints to the new limits. To counteract any excursions driven by external disturbances, we implement an active rejection mechanism: whenever the measured knee angle violates the software limits, we (1) project the commanded target onto the nearest permissible bound and (2) elevate the proportional and derivative gains to $K_p = 60$ and $K_d = 1$, respectively, until the joint re-enters the safe region. This procedure enforces a soft joint-limit constraint exclusively in software—without altering hardware or contravening physical laws—while delivering high-gain corrective action against environmental perturbations.

E Additional Latent Space Analysis

In addition to the t-SNE analysis, we also apply Principal Component Analysis (PCA) [107] and Uniform Manifold Approximation and Projection (UMAP) [108] on the action latent vectors \bar{z}_{action} in Figure 8. Both PCA and UMAP projections reveal clear grouping according to the morphology class, with humanoid, quadruped, and hexapod embeddings forming distinct clusters. Compared to the t-SNE analysis, clusters about the topological, geometric, and kinematic variations are less pronounced and appear to be more cramped.

Furthermore, we show in Figure 9 the t-SNE analysis of the learned joint description latent space $f_\phi(d_j)$ for all joints from all embodiments in the GENBOT-1K dataset. Although the three morphologies still define the rough structure of this latent space, the learned embeddings for the joint descriptions seem to be much more entangled across the three morphology classes.

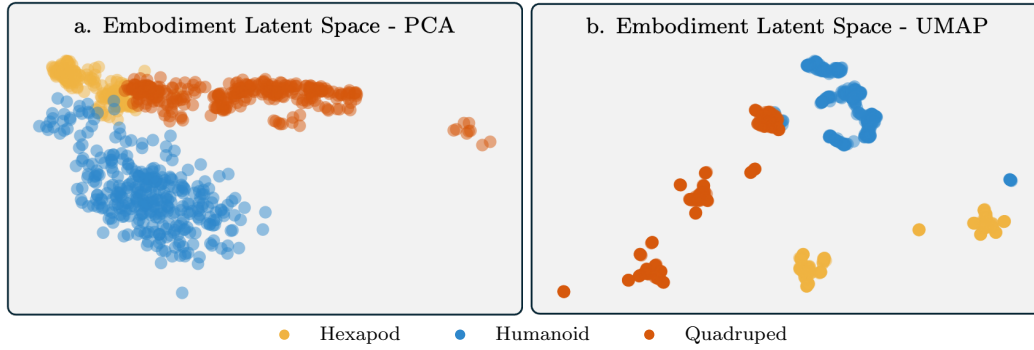


Figure 8: **Additional visualizations of the learned embodiment embeddings.** PCA (a.) and UMAP (b.) of the embodiment latent space (i.e., every point represents one robot, aggregated from all of its joint description vectors).

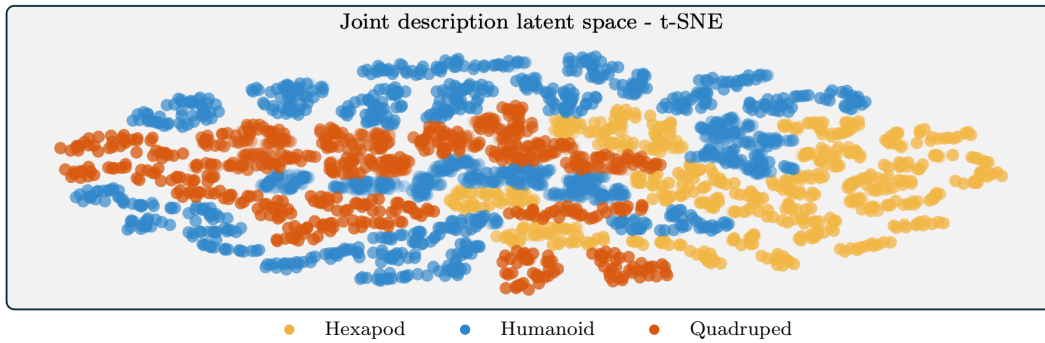


Figure 9: **Additional visualizations of the learned joint description embeddings.** t-SNE visualization of the joint description latent space of all joints from all embodiments in the GENBOT-1K dataset (i.e., every point represents one joint of a robot).

References

- [1] R. Nezafat, R. Shadmehr, and H. H. Holcomb. Long-term adaptation to dynamics of reaching movements: a PET study. *Experimental Brain Research*, 140(1):66–76, Sept. 2001.
- [2] D. Winter. *Biomechanics and Motor Control of Human Movement*, chapter 10, pages 250–280. John Wiley & Sons, New York, 2nd edition, 1990.
- [3] A. Maravita and A. Iriki. Tools for the body (schema). *Trends in Cognitive Sciences*, 8(2): 79–86, 2004. ISSN 1364-6613. doi:<https://doi.org/10.1016/j.tics.2003.12.008>. URL <https://www.sciencedirect.com/science/article/pii/S1364661303003450>.
- [4] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick. Segment anything. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3992–4003, 2023. doi:[10.1109/ICCV51070.2023.00371](https://doi.org/10.1109/ICCV51070.2023.00371).
- [5] B. Wen, W. Yang, J. Kautz, and S. Birchfield. Foundationpose: Unified 6d pose estimation and tracking of novel objects. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 17868–17879. IEEE, 2024. doi:[10.1109/CVPR52733.2024.01692](https://doi.org/10.1109/CVPR52733.2024.01692). URL <https://doi.org/10.1109/CVPR52733.2024.01692>.
- [6] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging properties in self-supervised vision transformers. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 9630–9640. IEEE, 2021. doi:[10.1109/ICCV48922.2021.00951](https://doi.org/10.1109/ICCV48922.2021.00951). URL <https://doi.org/10.1109/ICCV48922.2021.00951>.
- [7] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P. Huang, S. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jégou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski. Dinov2: Learning robust visual features without supervision. *Trans. Mach. Learn. Res.*, 2024, 2024. URL <https://openreview.net/forum?id=a68SUt6zFt>.
- [8] X. Zhai, A. Kolesnikov, N. Houlsby, and L. Beyer. Scaling vision transformers. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 1204–1213. IEEE, 2022. doi:[10.1109/CVPR52688.2022.01179](https://doi.org/10.1109/CVPR52688.2022.01179). URL <https://doi.org/10.1109/CVPR52688.2022.01179>.
- [9] C. Sun, A. Shrivastava, S. Singh, and A. Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 843–852. IEEE Computer Society, 2017. doi:[10.1109/ICCV.2017.97](https://doi.org/10.1109/ICCV.2017.97). URL <https://doi.org/10.1109/ICCV.2017.97>.
- [10] T. Tian, H. Li, B. Ai, X. Yuan, Z. Huang, and H. Su. Diffusion dynamics models with generative state estimation for cloth manipulation. *CoRR*, abs/2503.11999, 2025. doi:[10.48550/ARXIV.2503.11999](https://doi.org/10.48550/ARXIV.2503.11999). URL <https://doi.org/10.48550/ARXIV.2503.11999>.
- [11] D. Mahajan, R. B. Girshick, V. Ramanathan, K. He, M. Paluri, Y. Li, A. Bharambe, and L. van der Maaten. Exploring the limits of weakly supervised pretraining. In V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, editors, *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part II*, volume 11206 of *Lecture Notes in Computer Science*, pages 185–201. Springer, 2018. doi:[10.1007/978-3-030-01216-8_12](https://doi.org/10.1007/978-3-030-01216-8_12). URL https://doi.org/10.1007/978-3-030-01216-8_12.

- [12] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. F. Christiano, J. Leike, and R. Lowe. Training language models to follow instructions with human feedback. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/b1efde53be364a73914f58805a001731-Abstract-Conference.html.
- [13] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.
- [14] DeepSeek-AI, D. Guo, D. Yang, H. Zhang, J. Song, R. Zhang, R. Xu, Q. Zhu, S. Ma, P. Wang, X. Bi, X. Zhang, X. Yu, Y. Wu, Z. F. Wu, Z. Gou, Z. Shao, Z. Li, Z. Gao, A. Liu, B. Xue, B. Wang, B. Wu, B. Feng, C. Lu, C. Zhao, C. Deng, C. Zhang, C. Ruan, D. Dai, D. Chen, D. Ji, E. Li, F. Lin, F. Dai, F. Luo, G. Hao, G. Chen, G. Li, H. Zhang, H. Bao, H. Xu, H. Wang, H. Ding, H. Xin, H. Gao, H. Qu, H. Li, J. Guo, J. Li, J. Wang, J. Chen, J. Yuan, J. Qiu, J. Li, J. L. Cai, J. Ni, J. Liang, J. Chen, K. Dong, K. Hu, K. Gao, K. Guan, K. Huang, K. Yu, L. Wang, L. Zhang, L. Zhao, L. Wang, L. Zhang, L. Xu, L. Xia, M. Zhang, M. Zhang, M. Tang, M. Li, M. Wang, M. Li, N. Tian, P. Huang, P. Zhang, Q. Wang, Q. Chen, Q. Du, R. Ge, R. Zhang, R. Pan, R. Wang, R. J. Chen, R. L. Jin, R. Chen, S. Lu, S. Zhou, S. Chen, S. Ye, S. Wang, S. Yu, S. Zhou, S. Pan, S. S. Li, S. Zhou, S. Wu, S. Ye, T. Yun, T. Pei, T. Sun, T. Wang, W. Zeng, W. Zhao, W. Liu, W. Liang, W. Gao, W. Yu, W. Zhang, W. L. Xiao, W. An, X. Liu, X. Wang, X. Chen, X. Nie, X. Cheng, X. Liu, X. Xie, X. Liu, X. Yang, X. Li, X. Su, X. Lin, X. Q. Li, X. Jin, X. Shen, X. Chen, X. Sun, X. Wang, X. Song, X. Zhou, X. Wang, X. Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Y. Zhang, Y. Xu, Y. Li, Y. Zhao, Y. Sun, Y. Wang, Y. Yu, Y. Zhang, Y. Shi, Y. Xiong, Y. He, Y. Piao, Y. Wang, Y. Tan, Y. Ma, Y. Liu, Y. Guo, Y. Ou, Y. Wang, Y. Gong, Y. Zou, Y. He, Y. Xiong, Y. Luo, Y. You, Y. Liu, Y. Zhou, Y. X. Zhu, Y. Xu, Y. Huang, Y. Li, Y. Zheng, Y. Zhu, Y. Ma, Y. Tang, Y. Zha, Y. Yan, Z. Z. Ren, Z. Ren, Z. Sha, Z. Fu, Z. Xu, Z. Xie, Z. Zhang, Z. Hao, Z. Ma, Z. Yan, Z. Wu, Z. Gu, Z. Zhu, Z. Liu, Z. Li, Z. Xie, Z. Song, Z. Pan, Z. Huang, Z. Xu, Z. Zhang, and Z. Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL <https://arxiv.org/abs/2501.12948>.
- [15] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei. Scaling laws for neural language models. *CoRR*, abs/2001.08361, 2020. URL <https://arxiv.org/abs/2001.08361>.
- [16] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, and N. Fiedel. Palm: Scaling language modeling with pathways, 2022. URL <https://arxiv.org/abs/2204.02311>.
- [17] T. Gao, A. Fisch, and D. Chen. Making pre-trained language models better few-shot learners. In C. Zong, F. Xia, W. Li, and R. Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3816–3830, Online, Aug.

2021. Association for Computational Linguistics. doi:10.18653/v1/2021.acl-long.295. URL <https://aclanthology.org/2021.acl-long.295/>.
- [18] J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. de Las Casas, L. A. Hendricks, J. Welbl, A. Clark, T. Hennigan, E. Noland, K. Millikan, G. van den Driessche, B. Damoc, A. Guy, S. Osindero, K. Simonyan, E. Elsen, J. W. Rae, O. Vinyals, and L. Sifre. Training compute-optimal large language models. *CoRR*, abs/2203.15556, 2022. doi:10.48550/ARXIV.2203.15556. URL <https://doi.org/10.48550/arXiv.2203.15556>.
- [19] B. Ai, Y. Wang, Y. Tan, and S. Tan. Whodunit? learning to contrast for authorship attribution. In Y. He, H. Ji, Y. Liu, S. Li, C. Chang, S. Poria, C. Lin, W. L. Buntine, M. Liakata, H. Yan, Z. Yan, S. Ruder, X. Wan, M. Arana-Catania, Z. Wei, H. Huang, J. Wu, M. Day, P. Liu, and R. Xu, editors, *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2022 - Volume 1: Long Papers, Online Only, November 20-23, 2022*, pages 1142–1157. Association for Computational Linguistics, 2022. URL <https://aclanthology.org/2022.aacl-main.84>.
- [20] Z. Wu, B. Ai, and D. Hsu. Integrating common sense and planning with large language models for room tidying. In *RSS 2023 Workshop on Learning for Task and Motion Planning*, 2023. URL <https://openreview.net/forum?id=vuSI9mhDaBZ>.
- [21] Q. Gao, X. Pi, K. Liu, J. Chen, R. Yang, X. Huang, X. Fang, L. Sun, G. Kishore, B. Ai, S. Tao, M. Liu, J. Yang, C.-J. Lai, C. Jin, J. Xiang, B. Huang, D. Danks, H. Su, T. Shu, Z. Ma, L. Qin, and Z. Hu. Do vision-language models have internal world models? towards an atomic evaluation. In *ICLR 2025 Workshop on World Models: Understanding, Modelling and Scaling*, 2025. URL <https://openreview.net/forum?id=tpPv3ayoqo>.
- [22] K. Fang, P. Yin, A. Nair, H. Walke, G. Yan, and S. Levine. Generalization with lossy affordances: Leveraging broad offline data for learning visuomotor tasks. In K. Liu, D. Kulic, and J. Ichnowski, editors, *Conference on Robot Learning, CoRL 2022, 14-18 December 2022, Auckland, New Zealand*, volume 205 of *Proceedings of Machine Learning Research*, pages 106–117. PMLR, 2022. URL <https://proceedings.mlr.press/v205/fang23a.html>.
- [23] A. Kumar, A. Singh, F. D. Ebert, M. Nakamoto, Y. Yang, C. Finn, and S. Levine. Pre-training for robots: Offline RL enables learning new tasks in a handful of trials. In K. E. Bekris, K. Hauser, S. L. Herbert, and J. Yu, editors, *Robotics: Science and Systems XIX, Daegu, Republic of Korea, July 10-14, 2023*, 2023. doi:10.15607/RSS.2023.XIX.019. URL <https://doi.org/10.15607/RSS.2023.XIX.019>.
- [24] A. O’Neill, A. Rehman, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain, A. Tung, A. Bewley, A. Herzog, A. Irpan, A. Khazatsky, A. Rai, A. Gupta, A. Wang, A. Singh, A. Garg, A. Kembhavi, A. Xie, A. Brohan, A. Raffin, A. Sharma, A. Yavary, A. Jain, A. Balakrishna, A. Wahid, B. Burgess-Limerick, B. Kim, B. Schölkopf, B. Wulfe, B. Ichter, C. Lu, C. Xu, C. Le, C. Finn, C. Wang, C. Xu, C. Chi, C. Huang, C. Chan, C. Agia, C. Pan, C. Fu, C. Devin, D. Xu, D. Morton, D. Driess, D. Chen, D. Pathak, D. Shah, D. Büchler, D. Jayaraman, D. Kalashnikov, D. Sadigh, E. Johns, E. P. Foster, F. Liu, F. Ceola, F. Xia, F. Zhao, F. Stulp, G. Zhou, G. S. Sukhatme, G. Salhotra, G. Yan, G. Feng, G. Schiavi, G. Berseth, G. Kahn, G. Wang, H. Su, H. Fang, H. Shi, H. Bao, H. B. Amor, H. I. Christensen, H. Furuta, H. Walke, H. Fang, H. Ha, I. Mordatch, I. Radosavovic, I. Leal, J. Liang, J. Abou-Chakra, J. Kim, J. Drake, J. Peters, J. Schneider, J. Hsu, J. Bohg, J. Bingham, J. Wu, J. Gao, J. Hu, J. Wu, J. Wu, J. Sun, J. Luo, J. Gu, J. Tan, J. Oh, J. Wu, J. Lu, J. Yang, J. Malik, J. Silvério, J. Hejna, J. Booher, J. Thompson, J. Yang, J. Salvador, J. J. Lim, J. Han, K. Wang, K. Rao, K. Pertsch, K. Hausman, K. Go, K. Gopalakrishnan, K. Goldberg, K. Byrne, K. Oslund, K. Kawaharazuka, K. Black, K. Lin, K. Zhang, K. Ehsani, K. Lekkala, K. Ellis, K. Rana, K. Srinivasan, K. Fang, K. P. Singh,

- 620 K. Zeng, K. Hatch, K. Hsu, L. Itti, L. Y. Chen, L. Pinto, L. Fei-Fei, L. Tan, L. J. Fan, L. Ott,
621 L. Lee, L. Weihs, M. Chen, M. Lepert, M. Memmel, M. Tomizuka, M. Itkina, M. G. Cas-
622 tro, M. Spero, M. Du, M. Ahn, M. C. Yip, M. Zhang, M. Ding, M. Heo, M. K. Srirama,
623 M. Sharma, M. J. Kim, N. Kanazawa, N. Hansen, N. Heess, N. J. Joshi, N. Sünderhauf,
624 N. Liu, N. D. Palo, N. M. M. Shafiullah, O. Mees, O. Kroemer, O. Bastani, P. R. Sanketi,
625 P. T. Miller, P. Yin, P. Wohlhart, P. Xu, P. D. Fagan, P. Mitrano, P. Sermanet, P. Abbeel,
626 P. Sundaresan, Q. Chen, Q. Vuong, R. Rafailov, R. Tian, R. Doshi, R. Martín-Martín, R. Bai-
627 jal, R. Scalise, R. Hendrix, R. Lin, R. Qian, R. Zhang, R. Mendonca, R. Shah, R. Hoque,
628 R. Julian, S. Bustamante, S. Kirmani, S. Levine, S. Lin, S. Moore, S. Bahl, S. Dass, S. D.
629 Sonawani, S. Song, S. Xu, S. Haldar, S. Karamcheti, S. Adebola, S. Guist, S. Nasiriany,
630 S. Schaal, S. Welker, S. Tian, S. Ramamoorthy, S. Dasari, S. Belkhale, S. Park, S. Nair,
631 S. Mirchandani, T. Osa, T. Gupta, T. Harada, T. Matsushima, T. Xiao, T. Kollar, T. Yu,
632 T. Ding, T. Davchev, T. Z. Zhao, T. Armstrong, T. Darrell, T. Chung, V. Jain, V. Vanhoucke,
633 W. Zhan, W. Zhou, W. Burgard, X. Chen, X. Wang, X. Zhu, X. Geng, X. Liu, L. Xu, X. Li,
634 Y. Lu, Y. J. Ma, Y. Kim, Y. Chebotar, Y. Zhou, Y. Zhu, Y. Wu, Y. Xu, Y. Wang, Y. Bisk,
635 Y. Cho, Y. Lee, Y. Cui, Y. Cao, Y. Wu, Y. Tang, Y. Zhu, Y. Zhang, Y. Jiang, Y. Li, Y. Li,
636 Y. Iwasawa, Y. Matsuo, Z. Ma, Z. Xu, Z. J. Cui, Z. Zhang, and Z. Lin. Open x-embodiment:
637 Robotic learning datasets and RT-X models : Open x-embodiment collaboration. In *IEEE*
638 *International Conference on Robotics and Automation, ICRA 2024, Yokohama, Japan, May*
639 *13-17, 2024*, pages 6892–6903. IEEE, 2024. doi:10.1109/ICRA57147.2024.10611477. URL
640 <https://doi.org/10.1109/ICRA57147.2024.10611477>.
- 641 [25] D. Ghosh, H. R. Walke, K. Pertsch, K. Black, O. Mees, S. Dasari, J. Hejna, T. Kreiman,
642 C. Xu, J. Luo, Y. L. Tan, L. Y. Chen, Q. Vuong, T. Xiao, P. R. Sanketi, D. Sadigh, C. Finn,
643 and S. Levine. Octo: An open-source generalist robot policy. In D. Kulic, G. Venture, K. E.
644 Bekris, and E. Coronado, editors, *Robotics: Science and Systems XX, Delft, The Netherlands,*
645 *July 15-19, 2024*, 2024. doi:10.15607/RSS.2024.XX.090. URL [https://doi.org/10.](https://doi.org/10.15607/RSS.2024.XX.090)
646 [15607/RSS.2024.XX.090](https://doi.org/10.15607/RSS.2024.XX.090).
- 647 [26] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan,
648 K. Hausman, A. Herzog, J. Hsu, J. Ibarz, B. Ichter, A. Irpan, T. Jackson, S. Jesmonth,
649 N. J. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, K. Lee, S. Levine, Y. Lu,
650 U. Malla, D. Manjunath, I. Mordatch, O. Nachum, C. Parada, J. Peralta, E. Perez, K. Pertsch,
651 J. Quiambao, K. Rao, M. S. Ryoo, G. Salazar, P. R. Sanketi, K. Sayed, J. Singh, S. Sontakke,
652 A. Stone, C. Tan, H. T. Tran, V. Vanhoucke, S. Vega, Q. Vuong, F. Xia, T. Xiao, P. Xu, S. Xu,
653 T. Yu, and B. Zitkovich. RT-1: robotics transformer for real-world control at scale. In K. E.
654 Bekris, K. Hauser, S. L. Herbert, and J. Yu, editors, *Robotics: Science and Systems XIX,*
655 *Daegu, Republic of Korea, July 10-14, 2023*, 2023. doi:10.15607/RSS.2023.XIX.025. URL
656 <https://doi.org/10.15607/RSS.2023.XIX.025>.
- 657 [27] B. Zitkovich, T. Yu, S. Xu, P. Xu, T. Xiao, F. Xia, J. Wu, P. Wohlhart, S. Welker, A. Wahid,
658 Q. Vuong, V. Vanhoucke, H. T. Tran, R. Soricut, A. Singh, J. Singh, P. Sermanet, P. R. Sanketi,
659 G. Salazar, M. S. Ryoo, K. Reymann, K. Rao, K. Pertsch, I. Mordatch, H. Michalewski, Y. Lu,
660 S. Levine, L. Lee, T. E. Lee, I. Leal, Y. Kuang, D. Kalashnikov, R. Julian, N. J. Joshi, A. Irpan,
661 B. Ichter, J. Hsu, A. Herzog, K. Hausman, K. Gopalakrishnan, C. Fu, P. Florence, C. Finn,
662 K. A. Dubey, D. Driess, T. Ding, K. M. Choromanski, X. Chen, Y. Chebotar, J. Carbajal,
663 N. Brown, A. Brohan, M. G. Arenas, and K. Han. RT-2: vision-language-action models
664 transfer web knowledge to robotic control. In J. Tan, M. Toussaint, and K. Darvish, editors,
665 *Conference on Robot Learning, CoRL 2023, 6-9 November 2023, Atlanta, GA, USA*, volume
666 229 of *Proceedings of Machine Learning Research*, pages 2165–2183. PMLR, 2023. URL
667 <https://proceedings.mlr.press/v229/zitkovich23a.html>.
- 668 [28] P. Intelligence, K. Black, N. Brown, J. Darpinian, K. Dhabalia, D. Driess, A. Esmail, M. Equi,
669 C. Finn, N. Fusai, M. Y. Galliker, D. Ghosh, L. Groom, K. Hausman, B. Ichter, S. Jakubczak,
670 T. Jones, L. Ke, D. LeBlanc, S. Levine, A. Li-Bell, M. Mothukuri, S. Nair, K. Pertsch, A. Z.
671 Ren, L. X. Shi, L. Smith, J. T. Springenberg, K. Stachowicz, J. Tanner, Q. Vuong, H. Walke,

- 672 A. Walling, H. Wang, L. Yu, and U. Zhilinsky. $\pi_{0.5}$: a vision-language-action model with
673 open-world generalization, 2025. URL <https://arxiv.org/abs/2504.16054>.
- 674 [29] H. Fang, H. Fang, Z. Tang, J. Liu, C. Wang, J. Wang, H. Zhu, and C. Lu. RH20T: A
675 comprehensive robotic dataset for learning diverse skills in one-shot. In *IEEE Interna-*
676 *tional Conference on Robotics and Automation, ICRA 2024, Yokohama, Japan, May 13-*
677 *17, 2024*, pages 653–660. IEEE, 2024. doi:10.1109/ICRA57147.2024.10611615. URL
678 <https://doi.org/10.1109/ICRA57147.2024.10611615>.
- 679 [30] F. Ebert, Y. Yang, K. Schmeckpeper, B. Bucher, G. Georgakis, K. Daniilidis, C. Finn, and
680 S. Levine. Bridge data: Boosting generalization of robotic skills with cross-domain datasets.
681 In K. Hauser, D. A. Shell, and S. Huang, editors, *Robotics: Science and Systems XVIII, New*
682 *York City, NY, USA, June 27 - July 1, 2022*, 2022. doi:10.15607/RSS.2022.XVIII.063. URL
683 <https://doi.org/10.15607/RSS.2022.XVIII.063>.
- 684 [31] H. R. Walke, K. Black, T. Z. Zhao, Q. Vuong, C. Zheng, P. Hansen-Estruch, A. W. He,
685 V. Myers, M. J. Kim, M. Du, A. Lee, K. Fang, C. Finn, and S. Levine. Bridgedata V2: A
686 dataset for robot learning at scale. In J. Tan, M. Toussaint, and K. Darvish, editors, *Conference*
687 *on Robot Learning, CoRL 2023, 6-9 November 2023, Atlanta, GA, USA*, volume 229 of
688 *Proceedings of Machine Learning Research*, pages 1723–1736. PMLR, 2023. URL <https://proceedings.mlr.press/v229/walke23a.html>.
689
- 690 [32] F. Lin, Y. Hu, P. Sheng, C. Wen, J. You, and Y. Gao. Data scaling laws in imitation learning
691 for robotic manipulation. *CoRR*, abs/2410.18647, 2024. doi:10.48550/ARXIV.2410.18647.
692 URL <https://doi.org/10.48550/arXiv.2410.18647>.
- 693 [33] H. Fang, C. Wang, M. Gou, and C. Lu. Graspnet-1billion: A large-scale benchmark for gen-
694 eral object grasping. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recog-*
695 *nition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 11441–11450. Computer
696 Vision Foundation / IEEE, 2020. doi:10.1109/CVPR42600.2020.01146. URL [https://](https://openaccess.thecvf.com/content_CVPR_2020/html/Fang_GraspNet-1Billion_A_Large-Scale_Benchmark_for_General_Object_Grasping_CVPR_2020_paper.html)
697 [openaccess.thecvf.com/content_CVPR_2020/html/Fang_GraspNet-1Billion_A_](https://openaccess.thecvf.com/content_CVPR_2020/html/Fang_GraspNet-1Billion_A_Large-Scale_Benchmark_for_General_Object_Grasping_CVPR_2020_paper.html)
698 [Large-Scale_Benchmark_for_General_Object_Grasping_CVPR_2020_paper.html](https://openaccess.thecvf.com/content_CVPR_2020/html/Fang_GraspNet-1Billion_A_Large-Scale_Benchmark_for_General_Object_Grasping_CVPR_2020_paper.html).
- 699 [34] W. Gao, B. Ai, J. Loo, Vinay, and D. Hsu. Intentionnet: Map-lite visual navigation at the
700 kilometre scale, 2024. URL <https://arxiv.org/abs/2407.03122>.
- 701 [35] B. Ai, Z. Wu, and D. Hsu. Invariance is key to generalization: Examining the role of repre-
702 sentation in sim-to-real transfer for visual navigation. In M. H. Ang Jr and O. Khatib, editors,
703 *Experimental Robotics*, pages 69–80, Cham, 2024. Springer Nature Switzerland. ISBN 978-
704 3-031-63596-0.
- 705 [36] B. Ai, W. Gao, Vinay, and D. Hsu. Deep visual navigation under partial observability. In *2022*
706 *International Conference on Robotics and Automation, ICRA 2022, Philadelphia, PA, USA,*
707 *May 23-27, 2022*, pages 9439–9446. IEEE, 2022. doi:10.1109/ICRA46639.2022.9811598.
708 URL <https://doi.org/10.1109/ICRA46639.2022.9811598>.
- 709 [37] A. Patel and S. Song. Get-zero: Graph embodiment transformer for zero-shot embodi-
710 ment generalization. *CoRR*, abs/2407.15002, 2024. doi:10.48550/ARXIV.2407.15002. URL
711 <https://doi.org/10.48550/arXiv.2407.15002>.
- 712 [38] A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, S. Dasari, S. Karamcheti, S. Nasiriany,
713 M. K. Srirama, L. Y. Chen, K. Ellis, P. D. Fagan, J. Hejna, M. Itkina, M. Lepert, Y. J. Ma,
714 P. T. Miller, J. Wu, S. Belkhale, S. Dass, H. Ha, A. Jain, A. Lee, Y. Lee, M. Memmel, S. Park,
715 I. Radosavovic, K. Wang, A. Zhan, K. Black, C. Chi, K. B. Hatch, S. Lin, J. Lu, J. Mercat,
716 A. Rehman, P. R. Sanketi, A. Sharma, C. Simpson, Q. Vuong, H. R. Walke, B. Wulfe, T. Xiao,
717 J. H. Yang, A. Yavary, T. Z. Zhao, C. Agia, R. Baijal, M. G. Castro, D. Chen, Q. Chen,
718 T. Chung, J. Drake, E. P. Foster, J. Gao, D. A. Herrera, M. Heo, K. Hsu, J. Hu, D. Jack-
719 son, C. Le, Y. Li, R. Lin, Z. Ma, A. Maddukuri, S. Mirchandani, D. Morton, T. Nguyen,

- 720 A. O'Neill, R. Scalise, D. Seale, V. Son, S. Tian, E. Tran, A. E. Wang, Y. Wu, A. Xie,
721 J. Yang, P. Yin, Y. Zhang, O. Bastani, G. Berseth, J. Bohg, K. Goldberg, A. Gupta, A. Gupta,
722 D. Jayaraman, J. J. Lim, J. Malik, R. Martín-Martín, S. Ramamoorthy, D. Sadigh, S. Song,
723 J. Wu, M. C. Yip, Y. Zhu, T. Kollar, S. Levine, and C. Finn. DROID: A large-scale in-the-
724 wild robot manipulation dataset. In D. Kulic, G. Venture, K. E. Bekris, and E. Coronado,
725 editors, *Robotics: Science and Systems XX, Delft, The Netherlands, July 15-19, 2024*, 2024.
726 doi:10.15607/RSS.2024.XX.120. URL <https://doi.org/10.15607/RSS.2024.XX.120>.
- 727 [39] N. Bohlinger, G. Czechmanowski, M. Krupka, P. Kicki, K. Walas, J. Peters, and D. Tateo. One
728 policy to run them all: an end-to-end learning approach to multi-embodiment locomotion.
729 *Conference on Robot Learning*, 2024.
- 730 [40] Z. Jia, X. Li, Z. Ling, S. Liu, Y. Wu, and H. Su. Improving policy optimization with
731 generalist-specialist learning. In K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvári, G. Niu,
732 and S. Sabato, editors, *International Conference on Machine Learning, ICML 2022, 17-23*
733 *July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning*
734 *Research*, pages 10104–10119. PMLR, 2022. URL <https://proceedings.mlr.press/v162/jia22a.html>.
735
- 736 [41] W. Wan, H. Geng, Y. Liu, Z. Shan, Y. Yang, L. Yi, and H. Wang. Unidexgrasp++: Improving
737 dexterous grasping policy learning via geometry-aware curriculum and iterative generalist-
738 specialist learning. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023,*
739 *Paris, France, October 1-6, 2023*, pages 3868–3879. IEEE, 2023. doi:10.1109/ICCV51070.
740 2023.00360. URL <https://doi.org/10.1109/ICCV51070.2023.00360>.
- 741 [42] R. Zhu, T. Dai, and O. Celiktutan. Cross domain policy transfer with effect cycle-consistency.
742 In *2024 IEEE International Conference on Robotics and Automation*. IEEE Explore, 2024.
- 743 [43] Y. Chen, Y. Chen, Z. Hu, T. Yang, C. Fan, Y. Yu, and J. Hao. Learning action-transferable
744 policy with action embedding. *arXiv preprint arXiv:1909.02291*, 2019.
- 745 [44] Y. Hu and G. Montana. Skill transfer in deep reinforcement learning under morphological
746 heterogeneity. *arXiv preprint arXiv:1908.05265*, 2019.
- 747 [45] X. Liu, D. Pathak, and D. Zhao. Meta-evolve: Continuous robot evolution for one-to-many
748 policy transfer. In *The Twelfth International Conference on Learning Representations*, 2024.
- 749 [46] T. Wang, R. Liao, J. Ba, and S. Fidler. Nervenet: Learning structured policy with graph neural
750 networks. In *International Conference on Learning Representations*, 2018.
- 751 [47] W. Huang, I. Mordatch, and D. Pathak. One policy to control them all: Shared mod-
752 ular policies for agent-agnostic control. In *Proceedings of the 37th International Con-*
753 *ference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119
754 of *Proceedings of Machine Learning Research*, pages 4455–4464. PMLR, 2020. URL
755 <http://proceedings.mlr.press/v119/huang20d.html>.
- 756 [48] B. Trabucco, M. Phielipp, and G. Berseth. Anymorph: Learning transferable policies by
757 inferring agent morphology. In *International Conference on Machine Learning*, pages 21677–
758 21691. PMLR, 2022.
- 759 [49] H. Furuta, Y. Iwasawa, Y. Matsuo, and S. S. Gu. A system for morphology-task generalization
760 via unified representation and behavior distillation. In *The Eleventh International Conference*
761 *on Learning Representations*, 2022.
- 762 [50] D. Shah, A. Sridhar, A. Bhorkar, N. Hirose, and S. Levine. Gnm: A general navigation model
763 to drive any robot. In *2023 IEEE International Conference on Robotics and Automation*
764 *(ICRA)*, pages 7226–7233. IEEE, 2023.

- [51] W. Song, H. Zhao, P. Ding, C. Cui, S. Lyu, Y. Fan, and D. Wang. Germ: A generalist robotic model with mixture-of-experts for quadruped robot. *arXiv preprint arXiv:2403.13358*, 2024.
- [52] R. Doshi, H. R. Walke, O. Mees, S. Dasari, and S. Levine. Scaling cross-embodied learning: One policy for manipulation, navigation, locomotion and aviation. In *8th Annual Conference on Robot Learning*, 2024.
- [53] M. Shafiee, G. Bellegarda, and A. Ijspeert. Manyquadrupeds: Learning a single locomotion policy for diverse quadruped robots. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3471–3477. IEEE, 2024.
- [54] A. Eftekhari, L. Weihs, R. Hendrix, E. Caglar, J. Salvador, A. Herrasti, W. Han, E. VanderBil, A. Kembhavi, A. Farhadi, et al. The one ring: a robotic indoor navigation generalist. *arXiv preprint arXiv:2412.14401*, 2024.
- [55] 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 K. Liu, D. Kulic, and J. Ichnowski, editors, *Conference on Robot Learning, CoRL 2022, 14-18 December 2022, Auckland, New Zealand*, volume 205 of *Proceedings of Machine Learning Research*, pages 1893–1903. PMLR, 2022. URL <https://proceedings.mlr.press/v205/feng23a.html>.
- [56] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.
- [57] T. Miki, J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter. Learning robust perceptive locomotion for quadrupedal robots in the wild. *Science Robotics*, 7(62):eabk2822, 2022.
- [58] G. B. Margolis and P. Agrawal. Walk these ways: Tuning robot control for generalization with multiplicity of behavior. In *Conference on Robot Learning*, pages 22–31. PMLR, 2023.
- [59] S. Choi, G. Ji, J. Park, H. Kim, J. Mun, J. H. Lee, and J. Hwangbo. Learning quadrupedal locomotion on deformable terrain. *Science Robotics*, 8(74):eade2256, 2023.
- [60] K. Caluwaerts, A. Iscen, J. C. Kew, W. Yu, T. Zhang, D. Freeman, K.-H. Lee, L. Lee, S. Saliceti, V. Zhuang, et al. Barkour: Benchmarking animal-level agility with quadruped robots. *arXiv preprint arXiv:2305.14654*, 2023.
- [61] M. Stasica, A. Bick, N. Bohlinger, O. Mohseni, J. Fritzsche, C. Hübner, J. Peters, and A. Seyfarth. Bridge the gap: Enhancing quadruped locomotion with vertical ground perturbations. In *Under review*, 2025. URL https://www.ias.informatik.tu-darmstadt.de/uploads/Team/NicoBohlinger/bridge_the_gap.pdf.
- [62] Z. Zhuang, Z. Fu, J. Wang, C. Atkeson, S. Schwertfeger, C. Finn, and H. Zhao. Robot parkour learning. In *Conference on Robot Learning (CoRL)*, 2023.
- [63] X. Cheng, K. Shi, A. Agarwal, and D. Pathak. Extreme parkour with legged robots. In *RoboLetics: Workshop on Robot Learning in Athletics@ CoRL 2023*, 2023.
- [64] J. Siekmann, K. Green, J. Warila, A. Fern, and J. Hurst. Blind bipedal stair traversal via sim-to-real reinforcement learning. In *Robotics: Science and Systems*, 2021.
- [65] A. Kumar, Z. Li, J. Zeng, D. Pathak, K. Sreenath, and J. Malik. Adapting rapid motor adaptation for bipedal robots. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1161–1168. IEEE, 2022.
- [66] I. Radosavovic, T. Xiao, B. Zhang, T. Darrell, J. Malik, and K. Sreenath. Real-world humanoid locomotion with reinforcement learning. *arXiv:2303.03381*, 2023.

- [67] Q. Liao, B. Zhang, X. Huang, X. Huang, Z. Li, and K. Sreenath. Berkeley humanoid: A research platform for learning-based control. *arXiv preprint arXiv:2407.21781*, 2024.
- [68] Z. Zhuang, S. Yao, and H. Zhao. Humanoid parkour learning. *arXiv preprint arXiv:2406.10759*, 2024.
- [69] E. Chane-Sane, J. Amigo, T. Flayols, L. Righetti, and N. Mansard. Soloparkour: Constrained reinforcement learning for visual locomotion from privileged experience. In *8th Annual Conference on Robot Learning*, 2024.
- [70] E. Kaufmann, L. Bauersfeld, A. Loquercio, M. Müller, V. Koltun, and D. Scaramuzza. Champion-level drone racing using deep reinforcement learning. *Nature*, 620(7976):982–987, 2023.
- [71] A. Kumar, Z. Fu, D. Pathak, and J. Malik. Rma: Rapid motor adaptation for legged robots. *Robotics: Science and Systems XVII*, 2021.
- [72] N. Rudin, D. Hoeller, P. Reist, and M. Hutter. Learning to walk in minutes using massively parallel deep reinforcement learning. In *Conference on Robot Learning*, pages 91–100. PMLR, 2022.
- [73] G. Margolis, G. Yang, K. Paigwar, T. Chen, and P. Agrawal. Rapid locomotion via reinforcement learning. In *Robotics: Science and Systems*, 2022.
- [74] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel. Sim-to-real transfer of robotic control with dynamics randomization. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 3803–3810. IEEE, 2018.
- [75] L. Campanaro, S. Gangapurwala, W. Merkt, and I. Havoutis. Learning and deploying robust locomotion policies with minimal dynamics randomization. *arXiv preprint arXiv:2209.12878*, 2022.
- [76] L. Smith, I. Kostrikov, and S. Levine. A walk in the park: Learning to walk in 20 minutes with model-free reinforcement learning. *arXiv preprint arXiv:2208.07860*, 2022.
- [77] L. Smith, Y. Cao, and S. Levine. Grow your limits: Continuous improvement with real-world rl for robotic locomotion. *arXiv preprint arXiv:2310.17634*, 2023.
- [78] J. Levy, T. Westenbroek, and D. Fridovich-Keil. Learning to walk from three minutes of real-world data with semi-structured dynamics models. In *8th Annual Conference on Robot Learning*, 2024.
- [79] N. Bohlinger, J. Kinzel, D. Palenicek, L. Antczak, and J. Peters. Gait in eight: Efficient on-robot learning for omnidirectional quadruped locomotion. *arXiv preprint arXiv:2503.08375*, 2025.
- [80] F. Jenelten, J. He, F. Farshidian, and M. Hutter. Dtc: Deep tracking control—a unifying approach to model-based planning and reinforcement-learning for versatile and robust locomotion. *arXiv preprint arXiv:2309.15462*, 2023.
- [81] M. Kasaei, M. Abreu, N. Lau, A. Pereira, and L. P. Reis. A cpg-based agile and versatile locomotion framework using proximal symmetry loss. *arXiv preprint arXiv:2103.00928*, 2021.
- [82] A. Zhao, J. Xu, M. Konaković-Luković, J. Hughes, A. Spielberg, D. Rus, and W. Matusik. Robogrammar: graph grammar for terrain-optimized robot design. *ACM Transactions on Graphics (TOG)*, 39(6):1–16, 2020.

- [83] T. Azakami, H. Kera, and K. Kawamoto. Adversarial body shape search for legged robots. In *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 682–687. IEEE, 2022.
- [84] C. Rajani, K. Arndt, D. Blanco-Mulero, K. S. Luck, and V. Kyrki. Co-imitation: learning design and behaviour by imitation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 6200–6208, 2023.
- [85] C. Hazard, N. Pollard, and S. Coros. Automated design of robotic hands for in-hand manipulation tasks. *International Journal of Humanoid Robotics*, 17(01):1950029, 2020.
- [86] A. Gupta, L. Fan, S. Ganguli, and L. Fei-Fei. Metamorph: Learning universal controllers with transformers. *arXiv preprint arXiv:2203.11931*, 2022.
- [87] A. Patel and S. Song. Get-zero: Graph embodiment transformer for zero-shot embodiment generalization. *arXiv preprint arXiv:2407.15002*, 2024.
- [88] X. Cheng, Y. Ji, J. Chen, R. Yang, G. Yang, and X. Wang. Expressive whole-body control for humanoid robots. In D. Kulic, G. Venture, K. E. Bekris, and E. Coronado, editors, *Robotics: Science and Systems XX, Delft, The Netherlands, July 15-19, 2024*, 2024. doi:10.15607/RSS.2024.XX.107. URL <https://doi.org/10.15607/RSS.2024.XX.107>.
- [89] J. Bjorck, F. Castañeda, N. Cherniadev, X. Da, R. Ding, Linxi, Y. Fang, D. Fox, F. Hu, S. Huang, J. Jang, Z. Jiang, J. Kautz, K. Kundalia, L. Lao, Z. Li, Z. Lin, K. Lin, G. Liu, E. LLontop, L. Magne, A. Mandlekar, A. Narayan, S. Nasiriany, S. Reed, Y. L. Tan, G. Wang, Z. Wang, J. Wang, Q. Wang, J. Xiang, Y. Xie, Y. Xu, Z. Xu, S. Ye, Z. Yu, A. Zhang, H. Zhang, Y. Zhao, R. Zheng, and Y. Zhu. GR00T N1: an open foundation model for generalist humanoid robots. *CoRR*, abs/2503.14734, 2025. doi:10.48550/ARXIV.2503.14734. URL <https://doi.org/10.48550/arXiv.2503.14734>.
- [90] M. Ji, X. Peng, F. Liu, J. Li, G. Yang, X. Cheng, and X. Wang. Exbody2: Advanced expressive humanoid whole-body control. *CoRR*, abs/2412.13196, 2024. doi:10.48550/ARXIV.2412.13196. URL <https://doi.org/10.48550/arXiv.2412.13196>.
- [91] C. Sferrazza, D. Huang, X. Lin, Y. Lee, and P. Abbeel. Humanoidbench: Simulated humanoid benchmark for whole-body locomotion and manipulation. In D. Kulic, G. Venture, K. E. Bekris, and E. Coronado, editors, *Robotics: Science and Systems XX, Delft, The Netherlands, July 15-19, 2024*, 2024. doi:10.15607/RSS.2024.XX.061. URL <https://doi.org/10.15607/RSS.2024.XX.061>.
- [92] H. Shi, W. Wang, S. Song, and C. K. Liu. Toddlerbot: Open-source ml-compatible humanoid platform for loco-manipulation, 2025. URL <https://arxiv.org/abs/2502.00893>.
- [93] M. Liu, Z. Chen, X. Cheng, Y. Ji, R. Qiu, R. Yang, and X. Wang. Visual whole-body control for legged loco-manipulation. In P. Agrawal, O. Kroemer, and W. Burgard, editors, *Conference on Robot Learning, 6-9 November 2024, Munich, Germany*, volume 270 of *Proceedings of Machine Learning Research*, pages 234–257. PMLR, 2024. URL <https://proceedings.mlr.press/v270/liu25b.html>.
- [94] R. Yang, M. Zhang, N. Hansen, H. Xu, and X. Wang. Learning vision-guided quadrupedal locomotion end-to-end with cross-modal transformers. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL <https://openreview.net/forum?id=nhnJ3oo6AB>.
- [95] T. He, C. Zhang, W. Xiao, G. He, C. Liu, and G. Shi. Agile but safe: Learning collision-free high-speed legged locomotion. In D. Kulic, G. Venture, K. E. Bekris, and E. Coronado, editors, *Robotics: Science and Systems XX, Delft, The Netherlands, July 15-19, 2024*, 2024. doi:10.15607/RSS.2024.XX.059. URL <https://doi.org/10.15607/RSS.2024.XX.059>.

- [96] G. B. Margolis, G. Yang, K. Paigwar, T. Chen, and P. Agrawal. Rapid locomotion via reinforcement learning. *Int. J. Robotics Res.*, 43(4):572–587, 2024. doi:10.1177/02783649231224053. URL <https://doi.org/10.1177/02783649231224053>.
- [97] J. Tan, T. Zhang, E. Coumans, A. Iscen, Y. Bai, D. Hafner, S. Bohez, and V. Vanhoucke. Sim-to-real: Learning agile locomotion for quadruped robots. In H. Kress-Gazit, S. S. Srinivasa, T. Howard, and N. Atanasov, editors, *Robotics: Science and Systems XIV, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA, June 26-30, 2018*, 2018. doi:10.15607/RSS.2018.XIV.010. URL <http://www.roboticsproceedings.org/rss14/p10.html>.
- [98] H. Zhang, Y. Liu, J. Zhao, J. Chen, and J. Yan. Development of a Bionic Hexapod Robot for Walking on Unstructured Terrain. *Journal of Bionic Engineering*, 11(2):176–187, June 2014. ISSN 2543-2141. doi:10.1016/S1672-6529(14)60041-X. URL [https://doi.org/10.1016/S1672-6529\(14\)60041-X](https://doi.org/10.1016/S1672-6529(14)60041-X).
- [99] Z. Zang, M. Kawawa-Beaudan, W. Yu, T. Zhang, and A. Zakhor. Perceptive Hexapod Legged Locomotion for Climbing Joist Environments. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2738–2745, Detroit, MI, USA, Oct. 2023. IEEE. ISBN 978-1-66549-190-7. doi:10.1109/IROS55552.2023.10341957. URL <https://ieeexplore.ieee.org/document/10341957/>.
- [100] T. Qu, D. Li, A. Zakhor, W. Yu, and T. Zhang. Versatile locomotion skills for hexapod robots. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6885–6892, 2024. doi:10.1109/IROS58592.2024.10801714.
- [101] W. Ouyang, H. Chi, J. Pang, W. Liang, and Q. Ren. Adaptive locomotion control of a hexapod robot via bio-inspired learning. *Frontiers Neurobotics*, 15:627157, 2021. doi:10.3389/FNBOT.2021.627157. URL <https://doi.org/10.3389/fnbot.2021.627157>.
- [102] T. Azayev and K. Zimmerman. Blind Hexapod Locomotion in Complex Terrain with Gait Adaptation Using Deep Reinforcement Learning and Classification. *J Intell Robot Syst*, 2020.
- [103] J.-R. Chiu, Y.-C. Huang, H.-C. Chen, K.-Y. Tseng, and P.-C. Lin. Development of a Running Hexapod Robot with Differentiated Front and Hind Leg Morphology and Functionality. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3710–3717, Las Vegas, NV, USA, Oct. 2020. IEEE. doi:10.1109/iros45743.2020.9340811. URL <https://ieeexplore.ieee.org/document/9340811/>.
- [104] Haitao Yu, Wei Guo, Jing Deng, Mantian Li, and Hegao Cai. A CPG-based locomotion control architecture for hexapod robot. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5615–5621, Tokyo, Nov. 2013. IEEE. doi:10.1109/iros.2013.6697170. URL <http://ieeexplore.ieee.org/document/6697170/>.
- [105] M. Mittal, C. Yu, Q. Yu, J. Liu, N. Rudin, D. Hoeller, J. L. Yuan, R. Singh, Y. Guo, H. Mazhar, A. Mandlekar, B. Babich, G. State, M. Hutter, and A. Garg. Orbit: A unified simulation framework for interactive robot learning environments. *IEEE Robotics and Automation Letters*, 8(6):3740–3747, 2023. doi:10.1109/LRA.2023.3270034.
- [106] L. van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. URL <http://jmlr.org/papers/v9/vandermaaten08a.html>.
- [107] I. Jolliffe. *Principal component analysis*. Springer Verlag, New York, 2002.
- [108] L. McInnes and J. Healy. UMAP: uniform manifold approximation and projection for dimension reduction. *CoRR*, abs/1802.03426, 2018. URL <http://arxiv.org/abs/1802.03426>.

- 941 [109] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *7th International*
942 *Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.
943 OpenReview.net, 2019. URL <https://openreview.net/forum?id=Bkg6RiCqY7>.
- 944 [110] I. Loshchilov and F. Hutter. SGDR: stochastic gradient descent with warm restarts. In
945 *5th International Conference on Learning Representations, ICLR 2017, Toulon, France,*
946 *April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL <https://openreview.net/forum?id=Skq89Scxx>.
947