EGOHUMANOID-V2: HUMAN-TO-HUMANOID TRANSFER OF COORDINATED WHOLE-BODY SKILLS FOR LOCO-MANIPULATION

Anonymous authors
Paper under double-blind review

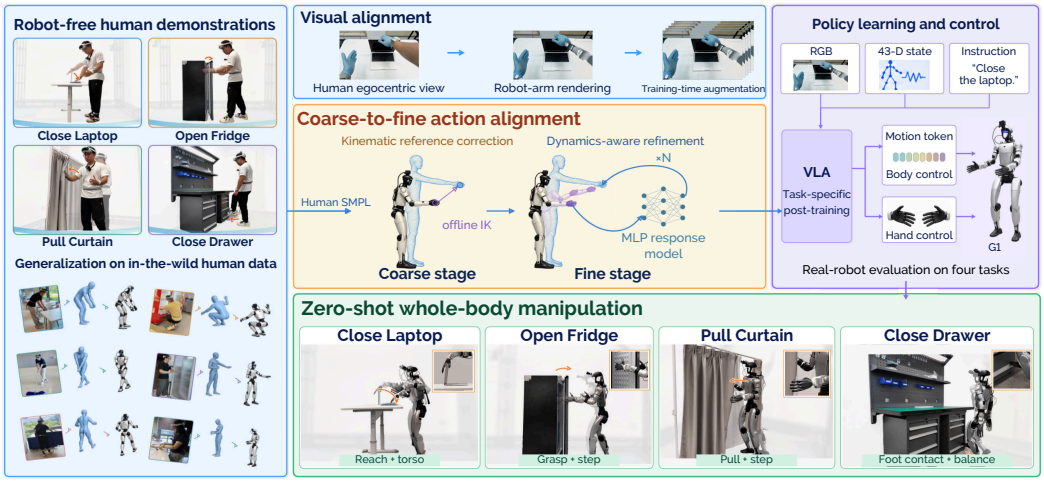


Figure 1: Introducing **EgoHumanoid-V2**, a human-to-humanoid skill transfer framework for whole-body loco-manipulation. Robot teleoperation is costly and hard to scale across environments, while human demonstrations capture diverse scenes and coordinated whole-body skills. Action and visual alignment bridge the embodiment gap for VLA training. Experiments on four real-world tasks show zero-shot skill transfer without target-task robot demonstrations.

ABSTRACT

Human demonstrations capture diverse scenes and rich whole-body skills without requiring robot teleoperation. Prior work on egocentric transfer has emphasized scene generalization in loco-manipulation under decoupled control, leaving direct transfer of coordinated whole-body skills less explored. We present **EgoHumanoid-V2**, a human-to-humanoid framework to transfer coordinated whole-body loco-manipulation skills from egocentric demonstrations. At its core, coarse-to-fine action alignment combines kinematic reference correction with dynamics-aware refinement. It improves end-effector pose accuracy while preserving whole-body coordination. We also use robot-arm rendering and training-time image augmentation to reduce the visual embodiment gap and improve viewpoint robustness. On four real-world tasks, vision-language-action (VLA) policies trained on aligned human data show zero-shot skill transfer without target-task robot demonstrations. Task scores are comparable to those of policies trained on teleoperation data at a lower collection cost. These results support human data as direct skill supervision. Demos are available at egohumanoid-v2-iclr2027.github.io.

1 INTRODUCTION

Human demonstrations offer rich manipulation experience across diverse scenes without robot teleoperation (Grauman et al., 2022; 2024). Yet the embodiment gap makes it challenging to exploit human motion for robot policy learning. Some approaches learn loco-manipulation priors from videos without action labels (Ye et al., 2025; Bu et al., 2025; Jiang et al., 2026; Ye et al., 2026). This

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

bypasses explicit action alignment, but may underutilize human motion (Lin et al., 2026). Recent work captures human hand trajectories with the Universal Manipulation Interface (UMI) (Chi et al., 2024; Liu et al., 2026) or VR motion capture (Yuan et al., 2026; Shi et al., 2026). Relative end-effector pose changes provide a shared human–robot action space (Kareer et al., 2026; Yuan et al., 2026). However, these methods primarily transfer bimanual manipulation and navigation, rather than coordinated whole-body loco-manipulation skills.

Humans naturally coordinate their arms, torso, and legs to extend their workspace and manipulate efficiently. Transferring these coordinated skills, however, requires more than matching end-effector trajectories. Whole-body tracking policies enable such coordination through joint whole-body control (Luo et al., 2026; Pan et al., 2026b). Unlike decoupled controllers that support arm inverse kinematics (IK) alongside separate balance and locomotion control (Ben et al., 2025; Li et al., 2025a), these policies jointly generate commands from full-body feedback. However, tracking human references alone can leave substantial end-effector error (Table 1). A straightforward approach is to correct the tracker's joint commands using online IK. Yet these corrections override part of the coordinated output without adjusting the remaining commands, potentially disrupting balance. Our implementation and evaluation confirm this risk (Appendix D). The challenge is therefore to improve interaction accuracy while preserving whole-body coordination.

We present **EgoHumanoid-V2**, a human-to-humanoid framework for transferring coordinated whole-body loco-manipulation skills from egocentric demonstrations (Fig. 1). Building on EgoHumanoid's exploration of egocentric human-to-humanoid transfer (Shi et al., 2026), we extend the scope from loco-manipulation under decoupled control to coordinated whole-body skills. At its core is offline, coarse-to-fine action alignment that accounts for the tracking policy's execution. In the coarse stage, kinematic reference correction uses offline IK to align motion references with human hand or foot targets, reducing geometric mismatch before tracking. In the fine stage, dynamics-aware refinement combines rollout feedback with a learned local response model to iteratively improve end-effector tracking. Together, these stages improve manipulation accuracy while preserving whole-body coordination. We also use robot-arm rendering to reduce the visual embodiment gap and training-time image augmentation to improve viewpoint robustness. The resulting observations and actions provide paired supervision for vision-language-action (VLA) post-training.

On four real-world tasks, policies trained on 200 human demonstrations per task achieve a mean score of 51.67% without target-task robot demonstrations. On two tasks, human collection requires 76.36% less labor per demonstration than teleoperation. Ablations examine action alignment through tracking accuracy and motion quality, and visual alignment through policy performance.

Our contributions are: **(a)** A human-to-humanoid framework for transferring coordinated whole-body loco-manipulation skills from egocentric demonstrations. **(b)** Coarse-to-fine action alignment through kinematic reference correction and dynamics-aware refinement. **(c)** Evidence of zero-shot skill transfer on four real-world tasks, with analyses of accuracy, policy scores, and collection cost.

2 Related Work

2.1 Whole-Body Control for Loco-Manipulation

Humanoid whole-body control supports imitation (Ji et al., 2025; Fu et al., 2025) and teleoperation (He et al., 2024; 2025b; Ze et al., 2025). For loco-manipulation, decoupled controllers combine arm joint targets or IK with adaptive lower-body control (Ben et al., 2025; Li et al., 2025a). Whole-body tracking policies instead use motion references and full-body feedback for joint control (Chen et al., 2026; Pan et al., 2026b), supporting diverse motions (Luo et al., 2026; Qi et al., 2026) and precise spatial tracking (Xu et al., 2026a). Retargeting improves reference quality (Araujo et al., 2026), interaction geometry (Yang et al., 2026b), and dynamic feasibility (Pan et al., 2026a). Execution errors nevertheless remain. Residual learning addresses dynamics mismatch (He et al., 2025a) and task interactions (Zhao et al., 2026). We instead correct references and refine execution offline under SONIC, keeping the controller fixed across tasks.

2.2 Learning from Human Demonstrations

Human demonstrations support robot learning through visual representations (Nair et al., 2023; Ma et al., 2023; Zeng et al., 2024), predictive pretraining (Wu et al., 2024), and motion priors from latent action models (Ye et al., 2025; Bu et al., 2025; Jiang et al., 2026) or world action models (Ye et al., 2026). Captured human trajectories provide direct action supervision (Hoque et al., 2026) through UMI-based interfaces (Chi et al., 2024; Zhaxizhuoma et al., 2025; Xu et al., 2025; Liu et al., 2026) or VR capture (Yuan et al., 2026). Wearable hand capture enables human–robot co-training (Kareer et al., 2025; Qiu et al., 2025; Tao et al., 2025), with domain adaptation improving transfer (Punamiya et al., 2025). Recent egocentric systems study scaling (Zheng et al., 2026) and humanoid control (Wei et al., 2026; Yang et al., 2026a). Loco-manipulation transfer extends to wheeled robots (Xu et al., 2026b) and humanoids with decoupled control (Shi et al., 2026; Zhong et al., 2026). Scene reconstruction supports contextual whole-body imitation (Allshire et al., 2025), while HuMI and BifrostUMI transfer coordinated manipulation (Nai et al., 2026; Wang et al., 2026). We instead refine action supervision offline under a fixed tracker, without task-specific training or fine-tuning.

3 Method

Figure 3 illustrates our action and visual alignment pipelines.

3.1 Problem Setup

For each task τ, human demonstrations contain synchronized egocentric images I, body motion x, and hand signals h, together with an instruction ℓ. We convert them into robot observations and actions to post-train a VLA policy π_θ^τ without target-task robot demonstrations. The policy predicts whole-body and hand actions from images, proprioceptive states, and instructions.

3.2 Data Collection and Policy Learning

We use a wearable system based on EgoHumanoid (Shi et al., 2026) to capture human demonstrations without robot hardware (Figure 2).

Human demonstration capture. A neck-mounted GoPro captures egocentric RGB images, while the PICO system captures body motion and hand poses. These streams are synchronized at 50 Hz. For deployment, we use a Unitree G1 with a head-mounted GoPro and BrainCo Revo2 hands. The human and robot cameras are placed at similar heights to reduce the viewpoint gap.

Training data construction. Offline action alignment converts human motion into SONIC tokens (Luo et al., 2026). Controller rollouts provide humanoid states for policy inputs and robot-arm rendering. The adapted images and aligned actions form training pairs, with image augmentation applied during training.

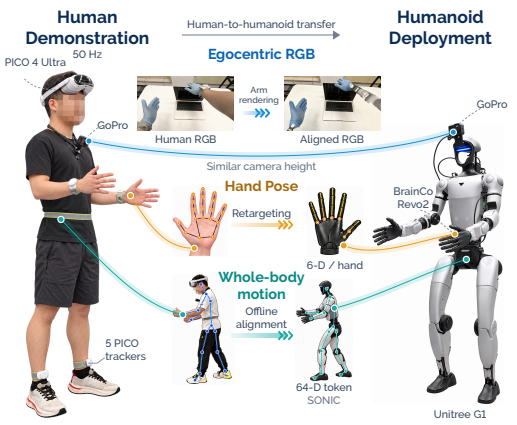


Figure 2: Hardware for human demonstration capture and humanoid deployment.

Policy learning and deployment. Using the aligned human demonstrations, we post-train one $\pi_{0.5}$ policy (Intelligence et al., 2025) per task without target-task robot demonstrations. The policy predicts action sequences from egocentric images, humanoid states, and language instructions, with a common interface for training and deployment:

$$o_t = (\widetilde{I}_t, q_t, \ell), \qquad a_t = (z_t, h_t^{\mathrm{L}}, h_t^{\mathrm{R}}), \tag{1}$$

where $\widetilde{I}_t$ denotes the adapted human image during training or the robot camera image at deployment, q_t is the 43-D G1 state, $z_t \in \mathbb{R}^{64}$ is a SONIC motion token, and $h_t^{\mathrm{L/R}} \in \mathbb{R}^6$ are Revo2 hand

commands. Controller rollouts supply q_t in the robot's state space during training. At deployment, SONIC executes motion tokens and the Revo2 driver executes hand commands. The same SONIC checkpoint serves alignment and deployment across all four tasks.

3.3 Action Alignment

Action alignment (Figure 3(a)) seeks accurate interaction trajectories while retaining the tracking policy's coordinated control. Starting from a controller rollout (H), the coarse stage corrects reference geometry (K), and the fine stage iteratively refines motion tokens using execution feedback (D).

Motion initialization. We obtain an initial robot motion by executing human body motion through SONIC in simulation, retaining the resulting joint trajectory $b^0_{1:T}$ and motion tokens, where $b_t \in \mathbb{R}^{29}$. We denote this initialization by H. Other motion references are also compatible with the framework; we compare their accuracy and motion quality in Table 2.

Target construction. We express motion in pelvis-centered frames with vertical axes fixed upright and horizontal axes following pelvis yaw. Human palm poses define the targets, with heights adjusted by the per-frame human–robot pelvis-height difference from the initial rollout. Targets remain fixed throughout refinement. Palm and sole target definitions appear in Appendix A.2.

3.3.1 Kinematic Reference Correction (K)

The coarse stage uses constrained offline IK to bring the initial robot reference closer to the human targets. The optimization reduces end-effector pose error while limiting changes to the original motion and maintaining temporal continuity. We encode the corrected reference $b^{\text{ref}}_{1:T}$ using SONIC's robot motion encoder $\mathcal{E}_r$ (Luo et al., 2026):

$$z^0_{1:T} = \mathcal{E}_r(b^{\text{ref}}_{1:T}). \tag{2}$$

Token replay measures the residual tracking errors corrected in the fine stage (Appendix A.2).

3.3.2 Dynamics-Aware Refinement (D)

The fine stage uses rollout feedback and an MLP response model $\mathcal{R}_\psi$. Paired perturbed and unperturbed rollouts train it to predict execution changes from kinematic changes (Appendix A.3).

At iteration k, a rollout of z^k provides the controller observation history s^k and end-effector residual r^k: position errors and, for hand targets, relative rotation vectors from executed poses to fixed targets. Holding the rollout context fixed, SONIC's control decoder $\mathcal{D}_c$ maps candidate tokens to joint commands. We predict their effect on execution as

$$\begin{aligned} \Delta u(z) &= \mathcal{D}_c(z, s^k) - \mathcal{D}_c(z^k, s^k), \\ \widehat{\Delta e}(z) &= \mathcal{R}_\psi\big(J^k \Delta u(z)\big)\,. \end{aligned} \tag{3}$$

Here, J^k maps joint-command changes to kinematic end-effector changes. We optimize the tokens so that the predicted response compensates for the current residual:

$$\mathcal{L}_{\text{track}}(z) = \|\widehat{\Delta e}(z) - r^k\|^2. \tag{4}$$

Gradients pass through the fixed response model and control decoder, not the simulator. Each updated sequence is evaluated in a fresh rollout, which supplies the residual and context for the next iteration. We select a validated sequence from the optimization history for policy training. The selected output, D_s, may come from an earlier round, including H or $H + K$. The full formulation adds temporal history, state and token conditioning, task weights, and regularization; selection criteria and implementation details appear in Appendices A.2–A.4.

3.4 Visual Alignment

Action alignment alone does not close the visual gap between human demonstrations and robot deployment (Table 4). We address differences in arm appearance and camera viewpoint through robot-arm rendering and training-time image augmentation (Figure 3(b)).

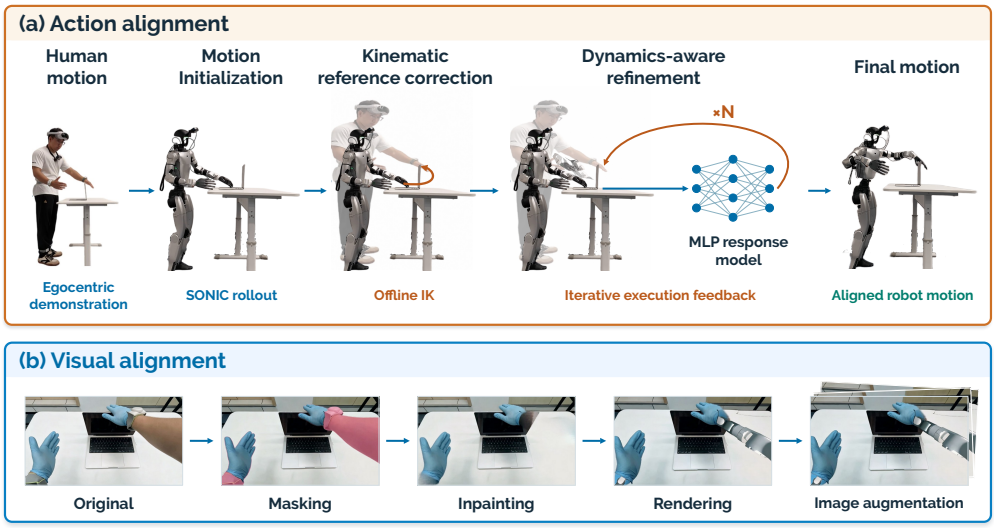


Figure 3: Action and visual alignment pipelines. (a) Human motion initializes a robot trajectory through a SONIC rollout (H). Offline IK corrects the reference geometry (K), and dynamics-aware refinement (D) uses an MLP response model and iterative execution feedback to produce aligned robot motion. (b) Human forearms and wrist trackers are masked and inpainted, then replaced with rendered robot arms. Image augmentation produces varied training views.

Robot-arm rendering. We adapt arm appearance (Chen et al., 2024; 2025). Matching blue gloves on humans during collection and robots during deployment simplify segmentation and visual alignment. We use SAM2 (Ravi et al., 2025) to segment human forearms and wrist-mounted PICO trackers, then fill the masked regions with DiffuEraser (Li et al., 2025b). We render G1 arms from aligned joint positions and adjust their translation, rotation, and uniform scale to match human wrists and forearms before compositing onto the inpainted backgrounds.

Image augmentation. Differences in movement amplitude and speed can alter camera viewpoints despite matching the initial camera height and angle. Random resized crops, small rotations, and photometric jitter improve robustness to these variations. At inference, the policy uses robot images directly (Appendix B.1). Table 4 evaluates rendering and augmentation separately and together.

4 Experiments

We evaluate skill transfer on a real humanoid and use simulation to examine the accuracy and motion quality of action alignment. We ask four questions:

Q1: Can human demonstrations teach whole-body loco-manipulation skills?

Q2: How do human and robot data compare in performance and cost?

Q3: How does action alignment improve accuracy?

Q4: How does visual alignment improve transfer?

4.1 Experimental Setup

We evaluate four coordinated whole-body manipulation tasks on Unitree G1, covering hand and foot interactions with posture adjustments or stepping (Figure 4).

(1) `Close Laptop`. The robot reaches above the laptop with its right hand and presses the lid shut, testing reaching accuracy and controlled downward contact until the lid is closed. (2) `Open Fridge`. The robot hooks its left hand into the door gap and pulls the door open, requiring precise hand placement and sustained contact as the robot adjusts its posture. (3) `Pull Curtain`. The robot grasps the curtain with its right hand and pulls it open by one

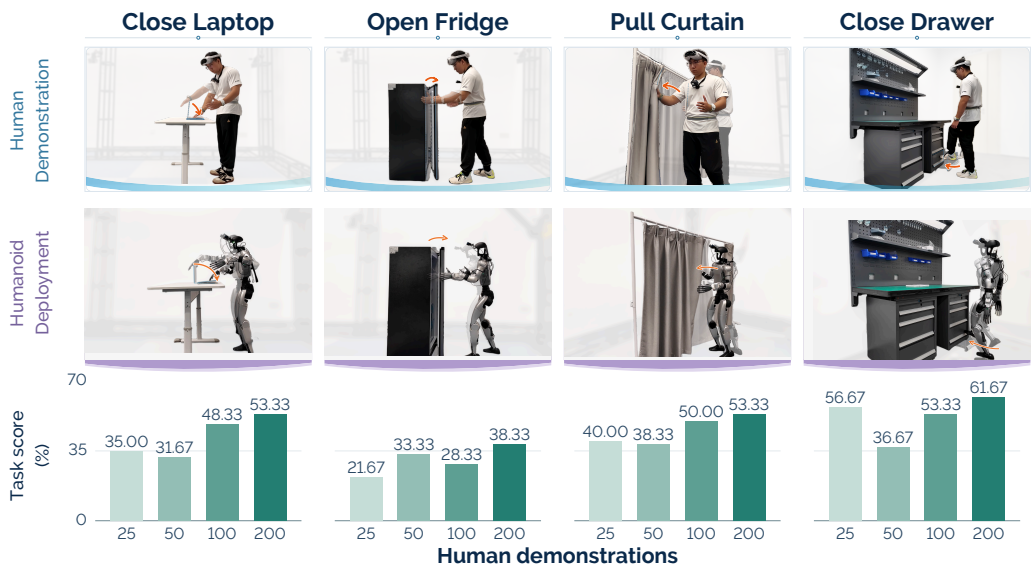


Figure 4: **Q1: Skill transfer on four real-world tasks.** Each task column shows human demonstrations (top), humanoid motion illustrations (middle), and task scores with 25, 50, 100, or 200 human demonstrations (bottom). Policies use no target-task robot demonstrations.

meter, combining grasp maintenance with arm and body motion over an extended pulling distance. (4) `Close Drawer`. The robot lifts its right foot and pushes the open drawer fully closed with its toe, requiring accurate foot placement while maintaining balance.

For each task, we post-train a separate $\pi_{0.5}$ policy for 30,000 steps with batch size 256, using the same pretrained checkpoint and training settings (Appendix B.1). Unless otherwise specified, human demonstrations undergo the full $H + K + D_s$ action-alignment pipeline, robot-arm rendering, and image augmentation; Close Drawer omits arm rendering. Human-data policies use no target-task robot demonstrations during post-training and are deployed directly on the robot.

Each policy is evaluated over 20 real-world trials with three sequential subtasks, each scored as success or failure. Averaging over subtasks and trials gives a task score that captures partial completion (Appendix B.3). Offline endpoint-error and motion-quality comparisons use a fixed random subset of 25 training trajectories per task (Appendices C.1–C.3). Cross-task means weight tasks equally.

4.2 Q1: Skill Transfer from Human Demonstrations

Figure 4 evaluates transfer with 25, 50, 100, and 200 human demonstrations per task.

With 200 human demonstrations per task, scores range from 38.33% on Open Fridge to 61.67% on Close Drawer, with a four-task mean of 51.67%. These results show that human demonstrations supervise both arm and foot manipulation through the same whole-body controller. Increasing the budget from 25 to 200 raises the mean from 38.34% to 51.67%, suggesting that additional human demonstrations improve transfer to robot deployment.

4.3 Q2: Human vs. Robot Demonstrations

Figure 5 compares human and robot demonstrations on Close Laptop and Pull Curtain under the same training and evaluation settings. Robot-data policies use 6, 12, 25, or 50 demonstrations. With 200 human demonstrations, both tasks reach 53.33%, compared with 63.33% and 56.67%, respectively, from 50 robot demonstrations. Human data approach robot-data performance but require more episodes, motivating a comparison of collection labor.

We measure collection labor in person-seconds, summing time across operators for capture, setup, calibration, troubleshooting, and scene resets. Shared overhead is amortized over episodes. Collect-

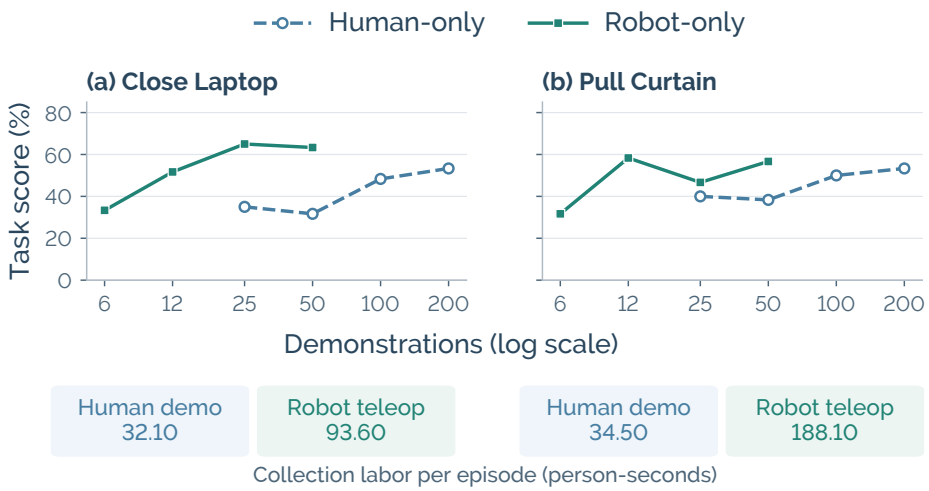


Figure 5: **Task scores and collection labor with human or robot demonstrations.** Each point represents an independently trained policy evaluated over 20 trials. Human-only results reuse Figure 4. Cost strips include capture, setup, calibration, troubleshooting, and resets.

ing a robot demonstration requires 2.92× as much labor as a human demonstration on Close Laptop and 5.45× on Pull Curtain. On average, human collection reduces labor per episode from 140.85 to 33.30 person-seconds, a 76.36% reduction.

On Close Laptop, 100 human demonstrations achieve 48.33% versus 63.33% from 50 robot demonstrations, using 31.41% less collection labor. On Pull Curtain, 200 human demonstrations reach 53.33% versus 56.67% from 50 robot demonstrations, using 26.63% less labor.

4.4 Q3: Action Alignment for Improved Accuracy

Why correct actions offline? We first examine whether online IK can improve endpoint accuracy during whole-body tracking. All three tested implementations produce unstable SONIC rollouts, including falls, across the four tasks, so we do not proceed to policy evaluation with these variants. Their designs and failure sequences are reported in Appendix D.

Kinematic reference correction and dynamics-aware refinement. Table 1 compares H, $H+K$, and the selected $H+K+D_s$ output under the same controller. Position error covers all four tasks; orientation error covers the three palm-interaction tasks. Drawer closing uses sole position only.

Table 1: **Q3: Effect of action alignment.** Top: position (cm) / orientation (°) errors in simulation (↓). Bottom: real-world task scores (%, ↑) with 200 demonstrations per task and identical visual preprocessing. Means are computed across tasks; Close Drawer uses sole position only and is excluded from orientation averages. Bold/underline mark best/second best. Selection follows Appendix C.2.

Alignment	Close Laptop Pos. (cm) / Ori. (°)	Open Fridge Pos. (cm) / Ori. (°)	Pull Curtain Pos. (cm) / Ori. (°)	Close Drawer Pos. (cm)	Mean Pos. (cm) / Ori. (°)
End-effector Tracking Error (↓)					
H	14.08 / 29.43	13.27 / 19.37	10.31 / 14.86	7.17	11.21 / 21.22
$H+K$	<u>4.22</u> / <u>13.84</u>	<u>4.24</u> / <u>14.72</u>	<u>3.13</u> / <u>12.74</u>	<u>6.22</u>	<u>4.45</u> / <u>13.77</u>
$H+K+D_s$	**3.04 / 9.50**	**2.76 / 5.77**	**2.02 / 4.85**	**5.80**	**3.41 / 6.71**
Real-World Task Score (%, ↑)					
H	0.00	0.00	28.33	56.67	21.25
$H+K$	<u>46.67</u>	<u>31.67</u>	<u>51.67</u>	<u>58.33</u>	<u>47.09</u>
$H+K+D_s$	**53.33**	**38.33**	**53.33**	**61.67**	**51.67**

Mean position error decreases from 11.21 cm at H to 4.45 cm after K and 3.41 cm at $H + K + D_s$; orientation error falls from 21.22° to 13.77° and 6.71°. Thus, K removes much of the initial position mismatch, while D further improves execution accuracy, especially in orientation. With 200 demonstrations per task, mean policy score likewise rises from 21.25% to 47.09% and 51.67%. The full pipeline improves policy scores over $H + K$ on every task. An expanded simulation study on 112 trajectories across 50 tasks shows the same stage-wise trend: position error falls from 17.03 to 5.22 to 3.83 cm (Appendix C.6, Table 8).

Motion initialization. Table 2 compares three motion initialization strategies: a SONIC rollout, GMR retargeting (Araujo et al., 2026), and whole-body IK adapted from HuMI (Nai et al., 2026). Each passes through the same subsequent K stage, eight refinement rounds, and selection procedure.

Table 2: **Effect of motion initialization.** Task-averaged endpoint error at D_s and whole-body motion quality. Pelvis drop measures maximum height loss; tilt, maximum base inclination; joint step, maximum frame-to-frame joint-angle change; extra yaw, heading deviation from the human reference. All variants share K, eight D rounds, and selection. Drawer orientation is excluded. Lower is better; bold/underline mark best/second best. Appendix C.3 defines the metrics; Tables 5 and 6 report results for each task and initialization.

	Endpoint Error		**Whole-Body Motion Quality**			
Motion initialization	Pos. (cm)	Ori. (°)	Pelvis drop (cm)	Tilt (°)	Joint step (°)	Extra yaw (°)
SONIC Rollout	**3.41**	6.71	**3.28**	**11.14**	**5.94**	**7.18**
Retargeting	<u>3.42</u>	**5.32**	<u>3.30</u>	<u>11.87</u>	<u>6.51</u>	<u>11.45</u>
Whole-Body IK	3.82	<u>6.35</u>	5.25	12.77	7.75	23.67

SONIC rollout and retargeting achieve similar position errors (3.41 versus 3.42 cm). Retargeting gives lower orientation error (5.32° versus 6.71°), while SONIC rollout reduces extra yaw from 11.45° to 7.18° and improves the other motion-quality metrics. This balance of accuracy and motion quality motivates our use of SONIC rollout for motion initialization.

Roles of K and D. Figure 6 supports one K round followed by multiple D rounds. Repeating K eight times changes position error only from 4.86 to 4.77 cm. Instead, adding eight D rounds after one K reduces position error to 3.97 cm and orientation error from 16.46° to 8.08°. Eight D rounds alone leave errors of 8.40 cm and 16.40°. Thus, K provides an accurate reference, while repeated D updates further reduce execution error; repeating K offers little additional benefit. The 50-task experiments likewise support one K round followed by multiple D rounds (Figure 7).

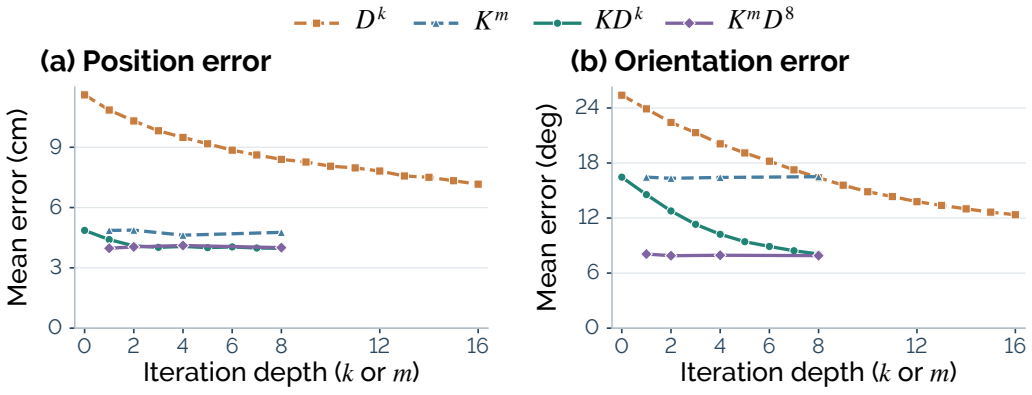


Figure 6: **Complementarity and depth of K and D.** Task-macro averages of raw stage outputs for D^k, repeated K^m, KD^k, and K^mD^8. Position averages four tasks; orientation averages the three hand tasks. Protocol and processing costs appear in Appendix C.4.

Learning the execution response. Table 3 compares the learned MLP with a unit response that equates kinematic and executed endpoint changes. Under identical settings, the MLP improves all four tasks: arm-task errors fall by 0.13–0.17 cm and 0.11–0.24°, while Close Drawer improves from 9.18 to 5.80 cm. The larger drawer gain suggests that modeling execution is especially useful when foot corrections require coordinated balance adjustments. On 50 tasks, raw KD^8 errors also improve from 4.25 to 4.10 cm and 9.39° to 9.21° (Table 9; training details in Appendix A.3).

Table 3: **MLP response versus unit response.** Endpoint position (cm) / orientation (°) error of historically selected outputs within an eight-round refinement budget; lower is better. Means weight tasks equally; orientation excludes drawer. Both variants share inputs and evaluation trajectories; bold marks the lower error for each task and the task-averaged results (Appendix C.5).

Response	Close Laptop Pos. (cm) / Ori. (°)	Open Fridge Pos. (cm) / Ori. (°)	Pull Curtain Pos. (cm) / Ori. (°)	Close Drawer Pos. (cm)	Mean Pos. (cm) / Ori. (°)
Unit response	3.18 / 9.74	2.89 / 5.88	2.19 / 5.06	9.18	4.36 / 6.89
MLP response	**3.04 / 9.50**	**2.76 / 5.77**	**2.02 / 4.85**	**5.80**	**3.41 / 6.71**

4.5 Q4: Visual Alignment for Easier Transfer

We evaluate robot-arm rendering and image augmentation separately and together, using 200 human demonstrations per task with identical action labels and training settings (Table 4). The baseline uses human images with standard model preprocessing only.

Table 4: **Q4: Visual-alignment ablation.** Task scores with 200 human demonstrations per task. Means average the available tasks; robot-arm rendering is not evaluated for foot-operated Close Drawer. Bold/underline mark best/second best; shading indicates the full pipeline.

Visual alignment		**Task score (%)**				**Mean**
Image augmentation	Robot-arm rendering	Close Laptop	Open Fridge	Pull Curtain	Close Drawer	
×	×	31.67	16.67	35.00	<u>50.00</u>	33.34
×	✓	51.67	23.33	<u>41.67</u>	—	38.89
✓	×	**61.67**	<u>33.33</u>	33.33	**61.67**	<u>47.50</u>
✓	✓	<u>53.33</u>	**38.33**	**53.33**	—	**48.33**

Across the three arm tasks, rendering, augmentation, and their combination achieve mean scores of 38.89%, 42.78%, and 48.33%, respectively, with unchanged action labels. Combining them gives the best results on Open Fridge (38.33%) and Pull Curtain (53.33%), while augmentation alone is best on Close Laptop (61.67%). On Pull Curtain, combining both components improves over rendering alone by 11.66 percentage points and augmentation alone by 20.00 points. Thus, their benefits are complementary but task-dependent.

5 Conclusion

EgoHumanoid-V2 transfers coordinated whole-body loco-manipulation skills from egocentric demonstrations through coarse-to-fine action alignment and visual adaptation. With 200 human demonstrations per task, VLA policies achieve a mean score of 51.67% across four real-world tasks without target-task robot demonstrations, demonstrating zero-shot skill transfer on the tested tasks. The results support human motion as direct supervision for coordinated humanoid skills.

Limitations and future work. Alignment can still produce unstable motions, and IK and repeated rollouts limit throughput. Stronger stability constraints and faster optimization remain directions for improving reliability and processing efficiency.

AI USE STATEMENT

Generative AI tools assisted with literature discovery and synthesis, method and experimental-design feedback, manuscript organization, drafting and editing, reference formatting, figure development, and software development. The authors reviewed this work and take responsibility for the final text, claims, citations, code, and artifacts.

REPRODUCIBILITY STATEMENT

Sections 3 and 4 describe the learning interface and controlled comparisons. Appendices A and B provide implementation and policy evaluation details. Appendices C and D report supplementary action-alignment experiments and online-IK diagnostics.

REFERENCES

Arthur Allshire, Hongsuk Choi, Junyi Zhang, David McAllister, Anthony Zhang, Chung Min Kim, Trevor Darrell, Pieter Abbeel, Jitendra Malik, and Angjoo Kanazawa. Visual imitation enables contextual humanoid control. In *CoRL*, pp. 794–815, 2025. URL https://proceedings.mlr.press/v305/allshire25a.html.

Joao Pedro Araujo, Yanjie Ze, Pei Xu, Jiajun Wu, and C. Karen Liu. Retargeting Matters: General motion retargeting for humanoid motion tracking. In *ICRA*, 2026. URL https://pei-xu.github.io/GMR/.

Qingwei Ben, Feiyu Jia, Jia Zeng, Junting Dong, Dahua Lin, and Jiangmiao Pang. HOMIE: Humanoid loco-manipulation with isomorphic exoskeleton cockpit. In *RSS*, 2025. doi: 10.15607/RSS.2025.XXI.070. URL https://www.roboticsproceedings.org/rss21/p070.html.

Qingwen Bu, Yanting Yang, Jisong Cai, Shenyuan Gao, Guanghui Ren, Maoqing Yao, Ping Luo, and Hongyang Li. Learning to act anywhere with task-centric latent actions. In *RSS*, 2025. doi: 10.15607/RSS.2025.XXI.014. URL https://www.roboticsproceedings.org/rss21/p014.html.

Lawrence Yunliang Chen, Kush Hari, Karthik Dharmarajan, Chenfeng Xu, Quan Vuong, and Ken Goldberg. Mirage: Cross-embodiment zero-shot policy transfer with cross-painting. In *RSS*, 2024.

Lawrence Yunliang Chen, Chenfeng Xu, Karthik Dharmarajan, Richard Cheng, Kurt Keutzer, Masayoshi Tomizuka, Quan Vuong, and Ken Goldberg. RoVi-Aug: Robot and viewpoint augmentation for cross-embodiment robot learning. In *CoRL*, pp. 209–233, 2025. URL https://proceedings.mlr.press/v270/chen25a.html.

Zixuan Chen, Mazeyu Ji, Xuxin Cheng, Xuanbin Peng, Xue Bin Peng, and Xiaolong Wang. GMT: General motion tracking for humanoid whole-body control. In *IROS*, 2026. URL https://gmt-humanoid.github.io/. To appear.

Cheng Chi, Zhenjia Xu, Chuer Pan, Eric Cousineau, Benjamin Burchfiel, Siyuan Feng, Russ Tedrake, and Shuran Song. Universal Manipulation Interface: In-the-wild robot teaching without in-the-wild robots. In *RSS*, 2024.

Zipeng Fu, Qingqing Zhao, Qi Wu, Gordon Wetzstein, and Chelsea Finn. HumanPlus: Humanoid shadowing and imitation from humans. In *CoRL*, pp. 2828–2844, 2025. URL https://proceedings.mlr.press/v270/fu25a.html.

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4D: Around the world in 3,000 hours of egocentric video. In *CVPR*, 2022.

Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, et al. Ego-Exo4D: Understanding skilled human activity from first- and third-person perspectives. In *CVPR*, 2024.

Tairan He, Zhengyi Luo, Wenli Xiao, Chong Zhang, Kris Kitani, Changliu Liu, and Guanya Shi. Learning human-to-humanoid real-time whole-body teleoperation. In *IROS*, 2024.

Tairan He, Jiawei Gao, Wenli Xiao, Yuanhang Zhang, Zi Wang, Jiashun Wang, Zhengyi Luo, Guanqi He, Nikhil Sobanbabu, Chaoyi Pan, Zeji Yi, Guannan Qu, Kris Kitani, Jessica Hodgins, Linxi "Jim" Fan, Yuke Zhu, Changliu Liu, and Guanya Shi. ASAP: Aligning simulation and real-world physics for learning agile humanoid whole-body skills. In *RSS*, 2025a. doi: 10.15607/RSS.2025.XXI.066. URL https://www.roboticsproceedings.org/rss21/p066.html.

Tairan He, Zhengyi Luo, Xialin He, Wenli Xiao, Chong Zhang, Weinan Zhang, Kris M Kitani, Changliu Liu, and Guanya Shi. OmniH2O: Universal and dexterous human-to-humanoid whole-body teleoperation and learning. In *CoRL*, pp. 1516–1540, 2025b. URL https://proceedings.mlr.press/v270/he25b.html.

Ryan Hoque, Peide Huang, David J Yoon, Mouli Sivapurapu, and Jian Zhang. EgoDex: Learning dexterous manipulation from large-scale egocentric video. In *ICLR*, 2026. URL https://openreview.net/forum?id=FFxkFMU89E.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. $\pi_{0.5}$: a vision-language-action model with open-world generalization. In *CoRL*, 2025.

Mazeyu Ji, Xuanbin Peng, Fangchen Liu, Jialong Li, Ge Yang, Xuxin Cheng, and Xiaolong Wang. ExBody2: Advanced expressive humanoid whole-body control. In *RSS Workshop on Whole-Body Control and Bimanual Manipulation*, 2025. URL https://openreview.net/forum?id=rzG7xQPSCE.

Haoran Jiang, Jin Chen, Qingwen Bu, Li Chen, Modi Shi, Yanjie Zhang, Delong Li, Chuanzhe Suo, Chuang Wang, Zhihui Peng, et al. WholeBodyVLA: Towards unified latent vla for whole-body loco-manipulation control. In *ICLR*, 2026. URL https://openreview.net/forum?id=OCJmVjyzN7.

Simar Kareer, Dhruv Patel, Ryan Punamiya, Pranay Mathur, Shuo Cheng, Chen Wang, Judy Hoffman, and Danfei Xu. EgoMimic: Scaling imitation learning via egocentric video. In *ICRA*, 2025. doi: 10.1109/ICRA55743.2025.11127989. URL https://ieeexplore.ieee.org/document/11127989.

Simar Kareer, Karl Pertsch, James Darpinian, Judy Hoffman, Danfei Xu, Sergey Levine, Chelsea Finn, and Suraj Nair. Emergence of human to robot transfer in vision-language-action models. In *RSS*, 2026. doi: 10.15607/RSS.2026.XXII.072. URL https://www.roboticsproceedings.org/rss22/p072.html.

Jialong Li, Xuxin Cheng, Tianshu Huang, Shiqi Yang, Ri-Zhao Qiu, and Xiaolong Wang. AMO: Adaptive motion optimization for hyper-dexterous humanoid whole-body control. In *RSS*, 2025a. doi: 10.15607/RSS.2025.XXI.061. URL https://www.roboticsproceedings.org/rss21/p061.html.

Xiaowen Li, Haolan Xue, Peiran Ren, and Liefeng Bo. DiffuEraser: A diffusion model for video inpainting. *arXiv preprint arXiv:2501.10018*, 2025b. doi: 10.48550/arXiv.2501.10018. URL https://arxiv.org/abs/2501.10018.

Fanqi Lin, Kushal Arora, Jean Mercat, Haruki Nishimura, Paarth Shah, Chen Xu, Mengchao Zhang, Mark Zolotas, Maya Angeles, Owen Pfannenstiehl, Andrew Beaulieu, and Jose Barreiros. A systematic study of data modalities and strategies for co-training large behavior models for robot manipulation. In *RSS*, 2026. doi: 10.15607/RSS.2026.XXII.007. URL https://www.roboticsproceedings.org/rss22/p007.html.

Songming Liu, Bangguo Li, Kai Ma, Lingxuan Wu, Hengkai Tan, Xiao Ouyang, Hang Su, and Jun Zhu. RDT2: Exploring the scaling limit of UMI data towards zero-shot cross-embodiment generalization. *arXiv preprint arXiv:2602.03310*, 2026. URL https://arxiv.org/abs/2602.03310.

Zhengyi Luo, Ye Yuan, Tingwu Wang, Chenran Li, Sirui Chen, Fernando Castañeda, Zi-Ang Cao, Jiefeng Li, David Minor, Qingwei Ben, et al. SONIC: Supersizing motion tracking for natural humanoid whole-body control. *Science Robotics*, 2026. URL https://nvlabs.github.io/SONIC/. In press.

Yecheng Jason Ma, Shagun Sodhani, Dinesh Jayaraman, Osbert Bastani, Vikash Kumar, and Amy Zhang. VIP: Towards universal visual reward and representation via value-implicit pre-training. In *ICLR*, 2023.

Ruiqian Nai, Boyuan Zheng, Junming Zhao, Haodong Zhu, Sicong Dai, Zunhao Chen, Yihang Hu, Yingdong Hu, Tong Zhang, Chuan Wen, and Yang Gao. Humanoid manipulation interface: Humanoid whole-body manipulation from robot-free demonstrations. In *CoRL*, 2026. To appear.

Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3M: A universal visual representation for robot manipulation. In *CoRL*, pp. 892–909, 2023. URL https://proceedings.mlr.press/v205/nair23a.html.

Chaoyi Pan, Changhao Wang, Haozhi Qi, Zixi Liu, Homanga Bharadhwaj, Akash Sharma, Tingfan Wu, Guanya Shi, Jitendra Malik, and Francois Hogan. SPIDER: Scalable physics-informed dexterous retargeting. In *IROS*, 2026a. URL https://jc-bao.github.io/spider-project/. To appear.

Yixuan Pan, Ruoyi Qiao, Li Chen, Kashyap Chitta, Liang Pan, Haoguang Mai, Qingwen Bu, Cunyuan Zheng, Hao Zhao, Ping Luo, and Hongyang Li. Agility Meets Stability: Versatile humanoid control with heterogeneous data. In *ICRA*, 2026b.

Ryan Punamiya, Dhruv Patel, Patcharapong Aphiwetsa, Pranav Kuppili, Lawrence Y. Zhu, Simar Kareer, Judy Hoffman, and Danfei Xu. EgoBridge: Domain adaptation for generalizable imitation from egocentric human data. In *NeurIPS*, 2025. doi: 10.52202/085713-1569. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/434756a76c44e220d5f32d6e0b488c0c-Abstract-Conference.html.

Zekun Qi, Xuchuan Chen, Jilong Wang, Chenghuai Lin, Yunrui Lian, Wenyao Zhang, Xinqiang Yu, He Wang, and Li Yi. Humanoid generative pre-training for zero-shot motion tracking. In *CVPR*, pp. 20834–20844, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Qi_Humanoid_Generative_Pre-Training_for_Zero-Shot_Motion_Tracking_CVPR_2026_paper.html.

Ri-Zhao Qiu, Shiqi Yang, Xuxin Cheng, Chaitanya Chawla, Jialong Li, Tairan He, Ge Yan, David J Yoon, Ryan Hoque, Lars Paulsen, et al. Humanoid policy ~ human policy. In *CoRL*, pp. 2888–2906, 2025. URL https://proceedings.mlr.press/v305/qiu25a.html.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. SAM 2: Segment anything in images and videos. In *ICLR*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/45c1f6a8cbf2da59ebf2c802b4f742cd-Paper-Conference.pdf.

Modi Shi, Shijia Peng, Jin Chen, Haoran Jiang, Tianyu Li, Ping Luo, Di Huang, Hongyang Li, and Li Chen. Unlocking in-the-wild loco-manipulation with robot-free egocentric demonstration. In *RSS*, 2026. doi: 10.15607/RSS.2026.XXII.204. URL https://www.roboticsproceedings.org/rss22/p204.html.

Tony Tao, Mohan Kumar Srirama, Jason Jingzhou Liu, Kenneth Shaw, and Deepak Pathak. DexWild: Dexterous human interactions for in-the-wild robot policies. In *RSS*, 2025. doi: 10.15607/RSS.2025.XXI.075. URL https://www.roboticsproceedings.org/rss21/p075.html.

Hongwu Wang, Chenhao Yu, Youhao Hu, Jiachen Zhang, Yuanyuan Li, and Shaqi Luo. BifrostUMI: Bridging robot-free demonstrations and humanoid whole-body manipulation. *arXiv preprint arXiv:2605.03452*, 2026. URL https://arxiv.org/abs/2605.03452.

Songlin Wei, Hongyi Jing, Boqian Li, Zhenyu Zhao, Jiageng Mao, Zhenhao Ni, Sicheng He, Sheng Zang, Xiawei Liu, Kaidi Kang, Jie Liu, Weiduo Yuan, Marco Pavone, Di Huang, and Yue Wang. Ψ_0: An open foundation model towards universal humanoid loco-manipulation. In *RSS*, 2026. doi: 10.15607/RSS.2026.XXII.021. URL https://www.roboticsproceedings.org/rss22/p021.html.

Hongtao Wu, Ya Jing, Chilam Cheang, Guangzeng Chen, Jiafeng Xu, Xinghang Li, Minghuan Liu, Hang Li, and Tao Kong. Unleashing large-scale video generative pre-training for visual robot manipulation. In *ICLR*, 2024. URL https://openreview.net/forum?id=NxoFmGgWC9.

Mengda Xu, Han Zhang, Yifan Hou, Zhenjia Xu, Linxi Fan, Manuela Veloso, and Shuran Song. DexUMI: Using human hand as the universal manipulation interface for dexterous manipulation. In *CoRL*, pp. 437–459, 2025. URL https://proceedings.mlr.press/v305/xu25b.html.

Qingyao Xu, Sheng Yin, Zibo Zhou, Ya Zhang, Siheng Chen, and Yue Hu. GLoRI: Closed-loop whole-body tracking with global-local reference interaction for humanoid loco-manipulation. *arXiv preprint arXiv:2609.05994*, 2026a. URL https://arxiv.org/abs/2609.05994.

Xiaomeng Xu, Jisang Park, Han Zhang, Eric Cousineau, Aditya Bhat, Jose Barreiros, Dian Wang, Jeannette Bohg, and Shuran Song. HoMMI: Learning whole-body mobile manipulation from human demonstrations. In *RSS*, 2026b. doi: 10.15607/RSS.2026.XXII.205. URL https://www.roboticsproceedings.org/rss22/p205.html.

Haoran Yang, Jiacheng Bao, Yucheng Xin, Haoming Song, Yuyang Tian, Bin Zhao, Dong Wang, and Xuelong Li. ZeroWBC: Learning natural whole-body humanoid interaction from human egocentric data. *arXiv preprint arXiv:2603.09170*, 2026a. URL https://arxiv.org/abs/2603.09170.

Lujie Yang, Xiaoyu Huang, Zhen Wu, Angjoo Kanazawa, Pieter Abbeel, Carmelo Sferrazza, C Karen Liu, Rocky Duan, and Guanya Shi. OmniRetarget: Interaction-preserving data generation for humanoid whole-body loco-manipulation and scene interaction. In *ICRA*, 2026b. URL https://omniretarget.github.io/.

Seonghyeon Ye, Joel Jang, Byeongguk Jeon, Sejune Joo, Jianwei Yang, Baolin Peng, Ajay Mandlekar, Reuben Tan, Yu-Wei Chao, Bill Yuchen Lin, et al. Latent action pretraining from videos. In *ICLR*, 2025.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, Ayaan Malik, Kyungmin Lee, William Liang, Nadun Ranawaka, Jiasheng Gu, Yinzhen Xu, Guanzhi Wang, Fengyuan Hu, Avnish Narayan, Johan Bjorck, Jing Wang, Gwanghyun Kim, Dantong Niu, Ruijie Zheng, Yuqi Xie, Jimmy Wu, Qi Wang, Ryan Julian, Danfei Xu, Yilun Du, Yevgen Chebotar, Scott Reed, Jan Kautz, Yuke Zhu, Linxi Fan, and Joel Jang. World action models are zero-shot policies. In *CoRL*, 2026. URL https://dreamzero0.github.io/. To appear.

Chengbo Yuan, Rui Zhou, Mengzhen Liu, Yingdong Hu, Shengjie Wang, Li Yi, Chuan Wen, Shanghang Zhang, and Yang Gao. MotionTrans: Human vr data enable motion-level learning for robotic manipulation policies. In *ICRA*, 2026. URL https://motiontrans.github.io/.

Yanjie Ze, Zixuan Chen, João Pedro Araújo, Zi ang Cao, Xue Bin Peng, Jiajun Wu, and C. Karen Liu. TWIST: Teleoperated whole-body imitation system. In *CoRL*, pp. 2143–2154, 2025. URL https://proceedings.mlr.press/v305/ze25a.html.

Jia Zeng, Qingwen Bu, Bangjun Wang, Wenke Xia, Li Chen, Hao Dong, Haoming Song, Dong Wang, Di Hu, Ping Luo, et al. Learning manipulation by predicting interaction. In *RSS*, 2024. doi: 10.15607/RSS.2024.XX.123. URL https://www.roboticsproceedings.org/rss20/p123.html.

Siheng Zhao, Yanjie Ze, Yue Wang, C Karen Liu, Pieter Abbeel, Guanya Shi, and Rocky Duan. ResMimic: From general motion tracking to humanoid whole-body loco-manipulation via residual learning. In *ICRA*, 2026. URL https://resmimic.github.io/.

Zhaxizhuom Zhaxizhuoma, Kehui Liu, Chuyue Guan, Zhongjie Jia, Ziniu Wu, Xin Liu, Tianyu Wang, Shuai Liang, Pengan CHEN, Pingrui Zhang, et al. FastUMI: A scalable and hardware-independent universal manipulation interface with dataset. In *CoRL*, pp. 3069–3093, 2025. URL https://proceedings.mlr.press/v305/zhaxizhuoma25a.html.

Ruijie Zheng, Dantong Niu, Yuqi Xie, Jing Wang, Mengda Xu, Yunfan Jiang, Fernando Castaneda, Fengyuan Hu, You Liang Tan, Letian Fu, Trevor Darrell, Furong Huang, Yuke Zhu, Danfei Xu, and Linxi Fan. EgoScale: Scaling dexterous manipulation with diverse egocentric human data. In *CoRL*, 2026. URL https://research.nvidia.com/labs/gear/egoscale/. To appear.

Rui Zhong, Yizhe Sun, Junjie Wen, Jinming Li, Chuang Cheng, Wei Dai, Zhiwen Zeng, Huimin Lu, Yichen Zhu, and Yi Xu. HumanoidExo: Scalable whole-body humanoid manipulation via wearable exoskeleton. In *ICRA*, 2026. URL https://humanoid-exo.github.io/.

The appendix covers implementation (Appendix A), policy training and evaluation (Appendix B), supplementary action-alignment results (Appendix C), and online-IK diagnostics (Appendix D).

A Implementation Details

A.1 Data Preparation and Validation

GoPro RGB, PICO whole-body tracking, and hand-pose signals form synchronized 50 Hz image–trajectory pairs. Signal validation and heading calibration precede H, followed by K and D. Accepted replays provide states and actions paired with aligned images for VLA post-training.

Each training sample uses the 43-dimensional state and the 76-dimensional action interface in Eq. 1.

Validation checks PICO–GoPro timing consistency, video/frame parity, endpoint tracking, joint increments, base tilt, and pelvis displacement. Without an external pulse, timing consistency measures repeatability rather than absolute synchronization error. These checks assess training-data validity, not task success.

A.2 Action Alignment Details

Coordinate frames and target registration. A world-space point p_t^W is expressed in the local pelvis frame as

$$p_t^L = R_z(\psi_t)^\top (p_t^W - p_{\text{pelvis},t}^W), \tag{5}$$

where ψ_t is pelvis yaw. The frame has a vertical z axis and removes world translation and heading, but not body tilt. Human palm targets retain local horizontal coordinates and orientation. Their vertical coordinate is $y_{t,z} = p_{t,z}^{\mathrm{H},L} + h_t^{\mathrm{H}} - h_t^{\mathrm{R},0}$, where h_t^{H} and $h_t^{\mathrm{R},0}$ are pelvis heights above the floor in the human recording and initial robot generation. Registration remains fixed across rounds. Human-conditioned controller execution supplies the initial joint reference; independent token replay measures tracking error against the fixed human palm targets.

Foot-position targets use the midpoint of human ankle and foot keypoints as a sole-center proxy, not a measured contact point. We estimate an episode-level floor offset from the human recording and apply the same pelvis-height registration. Foot-only tasks omit the orientation objective.

Kinematic reference correction (K). Both K and D target bilateral hand poses by default; foot-only tasks such as Close Drawer target only foot positions. For hand tasks, K optimizes only the arms and waist. Both IK solvers constrain deviations from the initial reference.

Dynamics-aware refinement (D). Joint-preservation weights for the legs, waist, and arms are $(20, 20, 0.1)$ for hand tasks and $(0.1, 20, 20)$ for foot-only tasks. These soft penalties regularize proposed corrections without freezing body parts during execution.

Using the response prediction in Eq. 3, we weight the endpoint residual by the task metric Q_τ. The context c^k and observation history s^k remain fixed during the local solve. We optimize

$$z^{k+1} = \arg\min_{z \in \mathcal{N}(z^k)} \|\widehat{\Delta e}(z) - r^k\|_{Q_\tau}^2 + \mathcal{L}_{\text{reg}}(z, z^k). \tag{6}$$

The regularizer includes the joint-command penalty $\|\Delta u(z)\|_{W_\tau}^2$, token-change and temporal penalties, and an IK correction prior. Updates are restricted to $\mathcal{N}(z^k)$.

Each round locally linearizes endpoint geometry and uses 400 Adam steps through the frozen decoder and MLP response model. The default schedule is one K round, seven regular D rounds, and one conservative D round, each evaluated by a fresh rollout. Historical candidate selection follows separately.

A.3 Response Model and Training

Inputs and temporal context. The response predictor $\mathcal{R}_\psi$ in Eq. 3 includes causal history and conditions on fixed baseline features c^k, omitted from the main-text notation. This history is distinct

from initialization H. Position and rotation changes for both palms and soles form a 24-D input. The current input and five exponentially filtered copies (time constants: 0.02, 0.1, 0.5, 2, and 5 s) form a 144-D causal history, conditioned on 93 robot-state features and the 64-D reference token.

Response training data. We construct response supervision from the G1 motion trajectories in BONES-SEED[1]. We replay reference and perturbed token sequences from identical simulator states and controller histories at 50 Hz. Perturbations use smooth random changes and token-refinement directions. Inputs are Jacobian-based endpoint changes between commands decoded under the same baseline observation; targets are executed endpoint-pose differences one policy step later. Training uses 250 motions from 68 capture dates (345,080 frame–perturbation pairs); validation uses 13 motions from seven separate dates (20,420 pairs).

Architecture and optimization. A frozen linear response and an MLP residual jointly predict the 24-dimensional endpoint response. The MLP has two hidden layers of width 64 with SiLU activations. Subtracting its output at zero history enforces zero response when the intervention history is zero. We normalize inputs and targets using training-set statistics and minimize normalized mean squared error with a squared residual penalty of weight 1. Random perturbations receive 75% of the training weight; the two refinement-derived direction families share the remaining 25%. Within each family, capture dates and source motions are balanced. We use AdamW with learning rate 3×10^{-5}, weight decay 10^{-4}, batch size 512, and gradient clipping at norm 5. Training runs for at most 150 epochs, retaining the checkpoint with the lowest validation loss. During token optimization, model weights and baseline features remain fixed. Gradients traverse the candidate-dependent history, not the simulator.

Endpoint configurations. For palm tasks, we blend the learned response with the unit response using a learned-response weight of 0.15. For foot-only tasks, we use the learned position response without an orientation objective. Configurations are chosen through task-level development rollouts and fixed across episodes and refinement rounds.

A.4 Motion Selection and Validation

Training-data acceptance checks endpoint accuracy, execution consistency, and whole-body stability. The error thresholds are 5 cm for position and 15° for orientation where applicable. Pelvis-height drop must remain within 0.10 m and base tilt within 25°. These thresholds do not define scores.

Eligible historical candidates are ranked by worst, then mean, normalized endpoint error and independently replayed in that order. Selection may retain H or $H + K$; episodes with no passing candidate are rejected. Diagnostic selection and aggregation follow Appendix C.

Task profiles include refrigerator trajectory conditioning and curtain hand-command and sequence-end processing. All profiles reuse the same low-level checkpoint without target-task training or fine-tuning. This does not imply that its pretraining excluded similar motions.

B Policy Training and Evaluation

B.1 Post-Training Configuration

We initialize $\pi_{0.5}$ from `pi05_base` and post-train in bfloat16 for 30,000 steps with global batch size 256. We use AdamW, gradient clipping at 1.0, and weight decay 10^{-10}. A cosine schedule has 1,000 warmup steps, a peak learning rate of 2.5×10^{-5}, and a terminal rate of 2.5×10^{-6}. These settings are shared across controlled comparisons.

Visual preprocessing. Robot-arm rendering follows Sec. 3.4 offline. Post-training adds random resized crops, in-plane rotation, and photometric jitter. Deployment uses robot images with standard preprocessing only.

[1] https://huggingface.co/datasets/bones-studio/seed

B.2 Training Data and Controlled Comparisons

Human data use $H + K + D_s$, robot-arm rendering, and image augmentation unless ablated; foot-operated Close Drawer omits arm rendering. Q1 uses 25, 50, 100, or 200 human demonstrations per task; action and visual ablations use 200. Q2 trains separate human-data and robot-data policies at the budgets in Sec. 4.3. All conditions share the policy backbone, action interface, and evaluation protocol.

In Table 1, H and $H+K$ supply stage-specific action labels; the full variant uses validated historical candidates (Appendix A.4). Visual preprocessing and training settings are identical.

B.3 Task Scoring Criteria

Each task has three sequential binary-scored subtasks. Evaluation stops at the first failure, with remaining subtasks scored zero. Trial scores are 0, $1/3$, $2/3$, or 1; task scores average all 20 attempted trials per condition:

$$\text{Task score } (\%) = \frac{100}{20} \sum_{i=1}^{20} \left(\frac{1}{3} \sum_{j=1}^{3} s_{ij} \right), \qquad s_{ij} \in \{0, 1\}. \tag{7}$$

The subtasks are evaluated in the following order.

Close Laptop. (1) The right hand reaches a suitable position above the laptop lid; (2) the right hand makes an effective downward pressing motion on the lid; (3) the lid is fully closed.

Open Fridge. (1) The left hand approaches the door gap; (2) the left hand hooks into the gap; (3) the refrigerator door opens.

Pull Curtain. (1) The right hand grasps the curtain; (2) the hand pulls the curtain; (3) the curtain is pulled open by one meter.

Close Drawer. (1) The right foot lifts, with its toe positioned in front of the lowest drawer's outer face; (2) the right foot pushes the drawer inward; (3) the drawer is fully closed.

B.4 Result Aggregation

Task scores measure partial completion, not full-task success rates. Means weight tasks equally over the available scores; visual-ablation rows with arm rendering exclude Close Drawer. No error bars are reported. Policy trials are separate from endpoint-error evaluations.

Starting-pose and object variations test within-task execution by four task-specific policies, not a single policy's unseen-task generalization.

C Supplementary Action-Alignment Experiments

We supplement Sec. 4.4 with four-task protocols and diagnostics (Appendices C.1–C.5) and a separate 50-task simulation study (Appendix C.6).

C.1 Shared Protocol and Metrics

Four-task comparisons use the same fixed random sample of 25 training trajectories per task (Sec. 4.1), with matched targets, registration, controller checkpoint, reset states, and evaluation windows. They measure replay accuracy separately from autonomous policy performance.

Position error is measured in centimeters and orientation error in degrees. Errors are averaged over frames and the relevant endpoints within each trajectory, then over trajectories. Cross-task means weight tasks equally. Position includes both palms for the three hand tasks and both soles for Close Drawer. Orientation means include only the three hand tasks. Missing measurements are

Table 5: **Task-wise endpoint error for motion initialization.** Position/orientation errors at D_s are in cm/°, averaged over 25 episodes per task. Drawer orientation is not evaluated. Lower is better; bold and underline denote the best and second-best values for each metric.

Task	SONIC Rollout	Retargeting	Whole-Body IK
Close Laptop	3.04 / 9.50	**2.67** / <u>6.33</u>	<u>2.75</u> / **6.19**
Open Fridge	**2.76** / <u>5.77</u>	<u>3.15</u> / **5.74**	3.68 / 7.25
Pull Curtain	**2.02** / <u>4.85</u>	<u>2.05</u> / **3.88**	2.73 / 5.61
Close Drawer	**5.80** / —	<u>5.80</u> / —	6.11 / —

not assigned numerical errors or replaced with a more favorable round. Reported metrics use two decimal places; averaging and ranking precede rounding.

C.2 Stage Comparison and Historical Selection

This subsection specifies selection for the offline error comparison in Table 1; policy-data validation follows Appendix A.4. The three hand tasks rank historical candidates using

$$S = 0.75\,\frac{e_{\text{pos}}}{5\text{ cm}} + 0.25\,\frac{e_{\text{ori}}}{15^\circ}. \tag{8}$$

Here, e_{pos} and e_{ori} are mean bilateral palm errors from the same candidate. Close Drawer minimizes mean bilateral sole-position error. Candidates include H, $H+K$, and all recorded D rounds, so the selected $H+K+D_s$ output may retain H or $H+K$.

The endpoint-error block uses the recorded candidate measurements without an additional validation replay or training-data acceptance filtering.

C.3 Motion Initialization and Whole-Body Motion Quality

We compare SONIC Rollout with GMR retargeting and HuMI-style whole-body IK (Nai et al., 2026), adapted to our 29-joint model. All references undergo encoding and replay through the same SONIC checkpoint, then one K and eight D rounds. This isolates initialization, not the cited methods' full pipelines.

For each initializer, D_s is selected from D^1–D^8 after K. Hand tasks rank candidates by the worst normalized bilateral palm error, using position/5 cm and orientation/15°. Mean normalized error and iteration index break ties. Close Drawer uses the analogous bilateral sole-position rule without orientation. No training-data acceptance filter is applied. Table 2 reports task-averaged endpoint and motion-quality results; Tables 5 and 6 give the task-wise breakdowns.

Motion-quality metrics. Tables 2 and 6 report four complementary diagnostics. **Pelvis drop** is the maximum decrease in pelvis height from its initial value, in cm. **Tilt** is the maximum angle between the base's vertical axis and world vertical. **Joint step** is the largest absolute change in any of the 29 body-joint angles between consecutive replay frames, measuring abrupt joint motion rather than walking step length. **Extra yaw** compares robot and human heading changes. We unwrap each yaw trajectory and subtract its initial angle, yielding ψ^r and ψ^h. The metric is

$$e_{\text{yaw}} = \max\left\{\left|\min_t \psi_t^r - \min_t \psi_t^h\right|,\ \left|\max_t \psi_t^r - \max_t \psi_t^h\right|,\ \left|\psi_T^r - \psi_T^h\right|\right\}. \tag{9}$$

This captures excessive or insufficient rotation at signed turning extrema and final heading. Angular metrics use degrees. Diagnostics are computed per trajectory, averaged within tasks, then equally across tasks.

C.4 Refinement Depth and Processing Cost

In Figure 6, $K^m D^k$ denotes m kinematic reference correction rounds followed by k dynamics-aware refinement rounds, starting from the SONIC Rollout reference. We compare D^8 with KD^8 and extend the D-only branch to 16 rounds. Repeated-K comparisons use $m \in \{1, 2, 4, 8\}$, both alone

Table 6: **Task-wise whole-body motion quality for motion initialization.** Metrics are averaged over the same fixed random 25-trajectory cohort per task as Table 2, using the definitions in Appendix C.3. Lower is better. Bold and underline mark the best and second-best values per task.

Task	Method	Pelvis drop (cm)	Tilt (°)	Joint step (°)	Extra yaw (°)
Close Laptop	SONIC Rollout	**0.28**	**9.62**	**2.87**	<u>3.94</u>
	Retargeting	<u>0.50</u>	<u>10.86</u>	<u>3.95</u>	**3.67**
	Whole-Body IK	2.06	11.32	6.07	22.60
Open Fridge	SONIC Rollout	<u>3.51</u>	**10.39**	**7.79**	**7.39**
	Retargeting	**3.35**	<u>11.62</u>	<u>8.28</u>	<u>13.96</u>
	Whole-Body IK	8.08	13.04	8.42	16.70
Pull Curtain	SONIC Rollout	**1.94**	**5.32**	<u>5.31</u>	<u>6.55</u>
	Retargeting	<u>2.48</u>	<u>6.16</u>	**5.26**	**5.66**
	Whole-Body IK	2.92	7.14	5.92	7.24
Close Drawer	SONIC Rollout	<u>7.40</u>	<u>19.22</u>	**7.78**	**10.84**
	Retargeting	**6.87**	**18.83**	<u>8.58</u>	<u>22.51</u>
	Whole-Body IK	7.95	19.57	10.60	48.16

Table 7: **Processing cost and endpoint accuracy.** Representative configurations from Figure 6. Time includes preparation, optimization, and replay, with reused H/K costs included. It excludes queueing, discarded failed attempts, and subsequent selection and validation. Values are equal-task means; orientation averages the three hand tasks. Errors use raw outputs at each depth (Figure 6).

Configuration	Time (s/frame)	Pos. (cm)	Ori. (°)
K^1	0.65	4.86	16.46
K^8	2.55	4.77	16.51
D^8	1.27	8.40	16.40
D^{16}	2.11	7.16	12.36
K^2D^8	1.80	4.04	7.90
K^4D^8	2.33	4.11	7.94
K^8D^8	3.38	4.01	7.90
K^1D^8 (Ours)	1.49	3.97	8.08

and followed by eight D rounds. The KD^k curve reports the raw output at each depth $k = 0, \ldots, 8$. These curves assess refinement depth without historical selection.

Table 7 pairs processing time with the endpoint errors in Figure 6. Elapsed times span job start to capture completion, including preparation, optimization, and replay. Reused H/K prefixes are charged at their recorded cost. Queueing, discarded failed attempts, and subsequent candidate selection and validation are excluded. Episode times are normalized by frame count, averaged within tasks, then averaged equally across tasks.

The timing runs use workstations with Ryzen 9 9950X3D CPUs and RTX 4090 GPUs (48 GB VRAM), and a machine with a Core Ultra 9 275HX CPU and an RTX 5090 Laptop GPU (24 GB VRAM). Each machine is configured for three episode workers; K/D rounds within an episode remain sequential. Times are pooled across these hosts without division by worker count. They characterize the logged processing runs, not a single-machine latency benchmark. K^1D^8 costs 1.49 s/frame; increasing K depth to eight raises this to 3.38 s/frame with similar endpoint error.

C.5 Response-Model Ablation

Table 3 compares the learned response (Appendix A.3) with a unit response that retains the Jacobian and token optimizer but omits learned corrections. Both share H/K inputs, targets, refinement settings, and eight D rounds; endpoint configurations are fixed across episodes. Position and orien-

tation errors come from the same historically selected candidate, unlike the raw outputs in Figure 6 (Appendix C.4).

C.6 Action Alignment across 50 Tasks

Scope and protocol. Stage, refinement-depth, and response comparisons share 112 paired trajectories across 50 tasks, separate from the four-task cohort, with equal task weights. This simulation study evaluates motion alignment, not autonomous policies. Stage comparisons use historical selection; depth and response comparisons use outputs at the specified round.

Kinematic correction and dynamics-aware refinement. Table 8 extends the stage-wise gains in Table 1: K reduces position/orientation errors from 17.03 cm/36.06° to 5.22 cm/21.81°; refinement and historical selection further reduce them to 3.83 cm/10.85°.

Table 8: **Stage comparison on the expanded benchmark.** Task-equal endpoint errors over 112 paired trajectories across 50 tasks; lower is better. $H + K + D_s$ uses historical candidate selection.

Stage	Pos. (cm)	Ori. (°)
H	17.03	36.06
$H + K$	5.22	21.81
$H + K + D_s$	**3.83**	**10.85**

Refinement depth. Figure 7 shows the raw stage outputs at matched refinement depths. Increasing K depth from one to eight changes position error from 5.22 to 5.01 cm. One K round followed by eight D rounds reaches 4.10 cm and 9.21°, compared with 5.22 cm and 21.81° after K alone. Eight D rounds without K yield 14.16 cm and 21.91°. As in Figure 6, repeated K offers little gain, while K followed by D outperforms either stage alone.

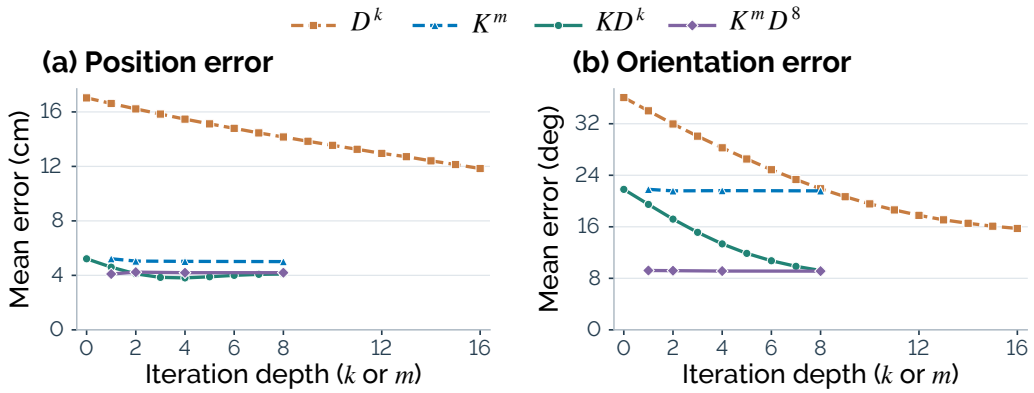


Figure 7: **Refinement depth on the expanded benchmark.** Task-equal averages of raw stage outputs for D^k, repeated K^m, KD^k, and $K^m D^8$ on paired trajectories.

Learned execution response. After one K and eight D rounds, the MLP lowers position error from 4.25 to 4.10 cm and orientation error from 9.39° to 9.21° (Table 9). These modest gains use matched-depth outputs, unlike the historical selections in Table 3.

D Online IK under a Whole-Body Tracking Controller

Online IK trades endpoint correction against coordinated balance. We test three SONIC command compositions on five trajectories per task: 60 runs across four tasks, separate from Appendix C. The IK objective includes position and orientation (orientation weight 0.01). Runs are neither accuracy-filtered nor retried for better outcomes.

Table 9: **Response-model comparison on the expanded benchmark.** Task-equal errors over 112 paired trajectories across 50 tasks after one K and eight D rounds, without historical selection; lower is better.

Response	Pos. (cm)	Ori. (°)
Unit response	4.25	9.39
MLP response	**4.10**	**9.21**

D.1 Three Command Compositions

For the three hand tasks, **A** assigns the arms and waist to IK while SONIC controls the legs. **B** adds SONIC's balance contribution to A. **C** restricts IK to the arms, leaving the waist and legs under SONIC control. For Close Drawer, A and B instead assign both legs to IK, without and with the balance contribution. C assigns only the right operating leg to IK; SONIC controls the remaining joints. The base and limbs remain dynamically coupled through balance and contact forces, even where SONIC commands are unchanged.

D.2 Accuracy and Stability

Table 10 averages tracking errors over frames, both endpoints, and episodes. Tilt follows the episode-maximum definition in Appendix C.3. Fall events can recur within a run and are not failed-episode counts. Initialization failures have no formal trajectory error and are counted separately.

Table 10: **Online-IK stability diagnostics on five episodes per task.** Errors use measured endpoints, not IK solutions. Init. falls counts failed runs; Events counts logged falls, including startup. Dashes indicate no formal replay. Bold/underline rank values within each task from lowest to second lowest; they do not indicate that execution is stable.

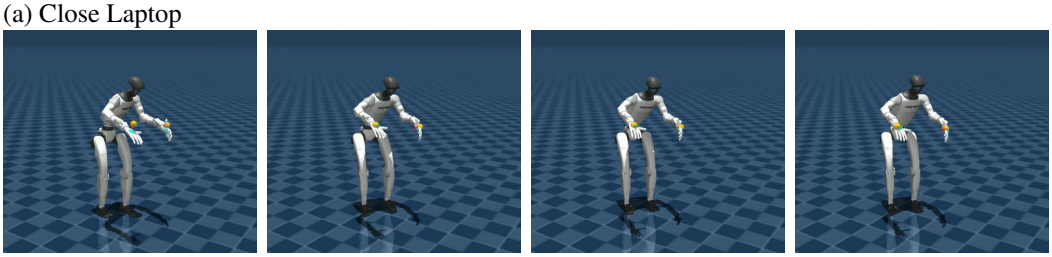
Variant	Task	Pos. (cm)	Ori. (°)	Tilt (°)	Init. falls	Events
A	Close Laptop	<u>4.45</u>	<u>10.43</u>	<u>16.97</u>	0/5	**0**
	Open Fridge	<u>2.33</u>	<u>15.34</u>	<u>18.97</u>	0/5	**0**
	Pull Curtain	<u>1.58</u>	<u>10.17</u>	<u>7.03</u>	0/5	**0**
	Close Drawer	—	—	—	5/5	—
B	Close Laptop	**2.43**	**8.28**	**13.30**	0/5	**0**
	Open Fridge	**1.55**	**14.45**	**17.26**	0/5	**0**
	Pull Curtain	**1.50**	**10.07**	**6.61**	0/5	**0**
	Close Drawer	—	—	—	5/5	—
C	Close Laptop	26.50	40.93	82.75	0/5	<u>9</u>
	Open Fridge	10.61	20.55	34.35	0/5	<u>1</u>
	Pull Curtain	9.00	18.66	10.87	0/5	**0**
	Close Drawer	—	—	—	5/5	—

Video inspection finds instability in all 60 runs. All 15 drawer runs fall during first-pose preparation; the 45 hand-task runs reach their final frames without stable tracking. Variant C logs nine laptop and one fridge fall events. Variant B remains visibly unstable despite 1.50–2.43 cm hand-task position error. We therefore exclude these variants from policy training. This finding is limited to the tested SONIC implementations.

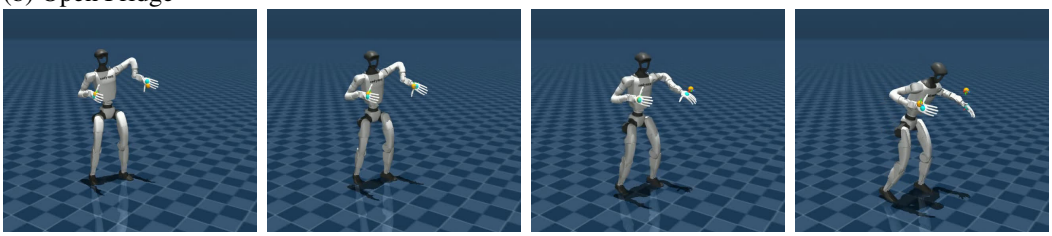
D.3 Qualitative Failure Sequences

Figures 8–10 show one simulation sequence per implementation and task, chosen by peak base tilt across the five runs, including initialization. These examples illustrate instability, not its frequency.

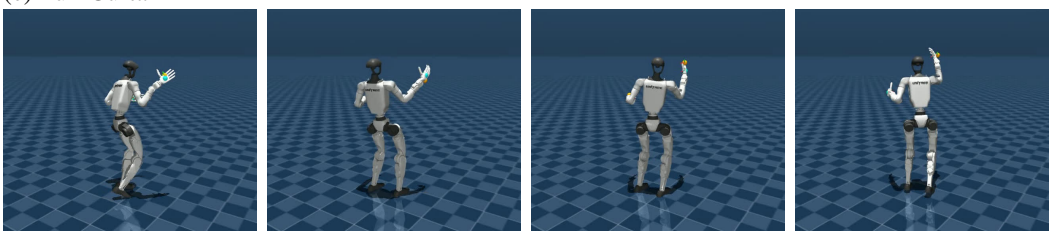
(a) Close Laptop



(b) Open Fridge



(c) Pull Curtain



(d) Close Drawer

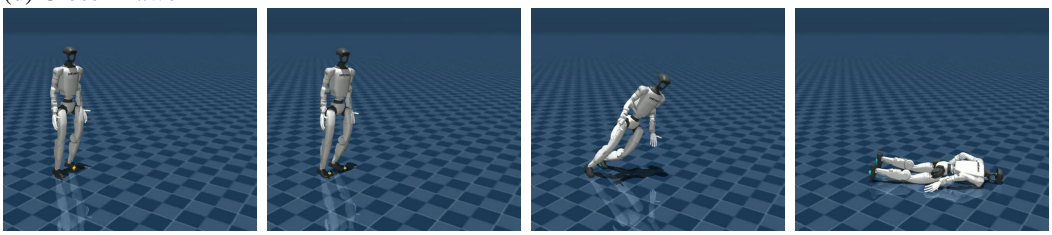
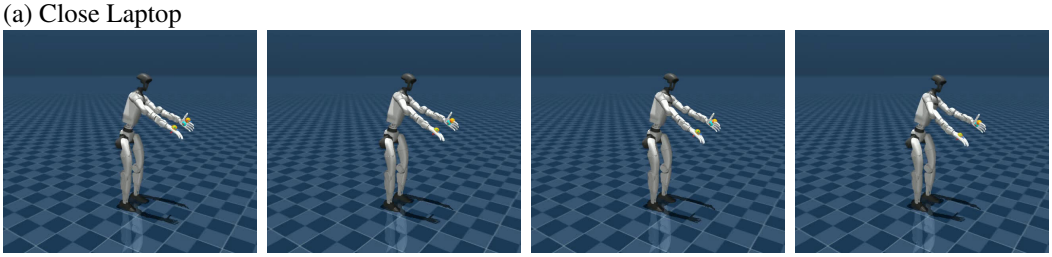
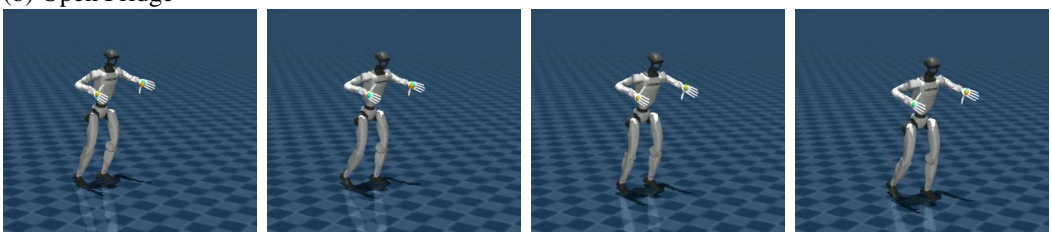


Figure 8: **Online-IK instability: variant A (IK without balance overlay).** Each row follows one continuous segment from left to right, selected by peak base tilt as described in Appendix D.3. A fixed crop within each sequence preserves relative motion and keeps robot scale comparable across tasks. Close Drawer ends in an initialization fall. The visualizations show recorded robot states without task objects; they do not depict autonomous policy evaluation.

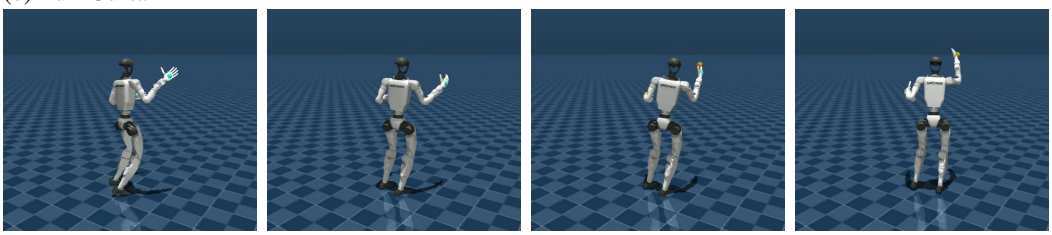
(a) Close Laptop



(b) Open Fridge



(c) Pull Curtain



(d) Close Drawer

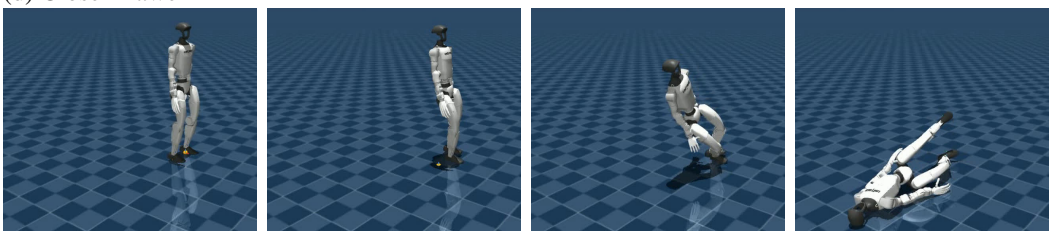
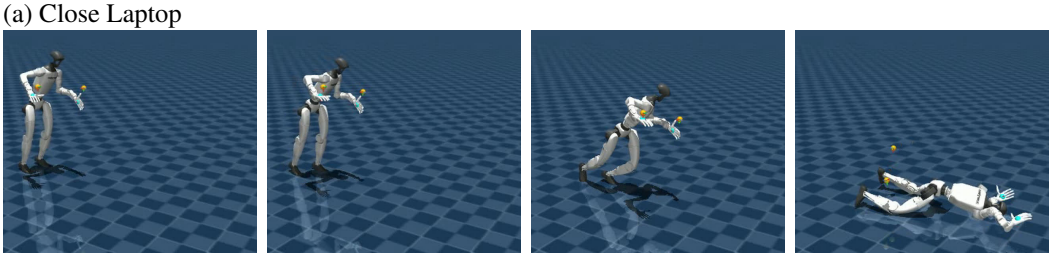
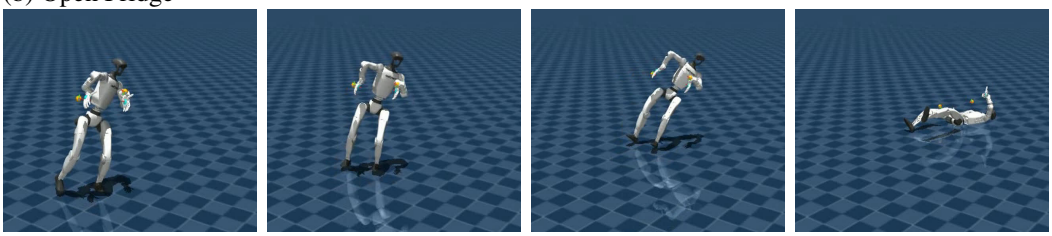


Figure 9: **Online-IK instability: variant B (IK with SONIC balance overlay).** Each row follows one continuous segment from left to right, selected by peak base tilt as described in Appendix D.3. A fixed crop within each sequence preserves relative motion and keeps robot scale comparable across tasks. Close Drawer ends in an initialization fall. The visualizations show recorded robot states without task objects; they do not depict autonomous policy evaluation.

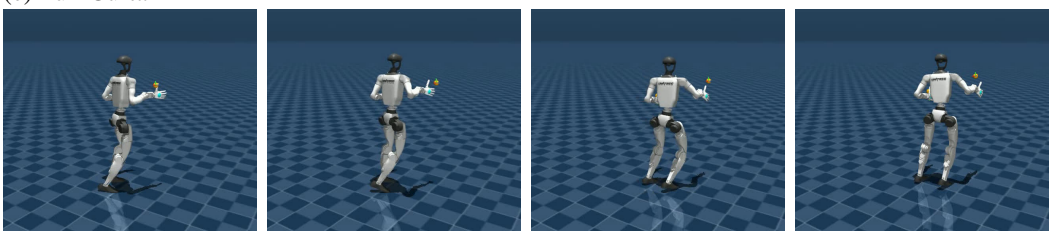
(a) Close Laptop



(b) Open Fridge



(c) Pull Curtain



(d) Close Drawer

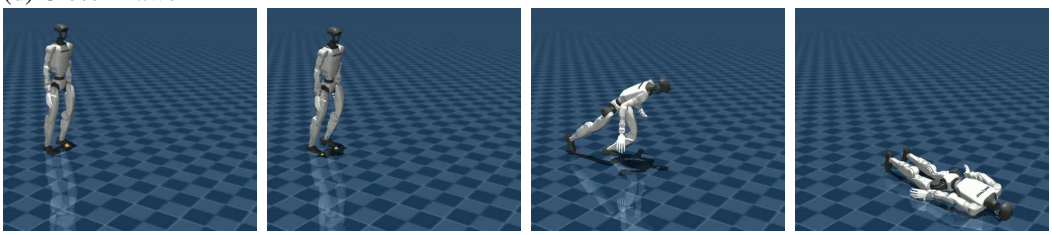


Figure 10: **Online-IK instability: variant C (restricted IK joint control).** Each row follows one continuous segment from left to right, selected by peak base tilt as described in Appendix D.3. A fixed crop within each sequence preserves relative motion and keeps robot scale comparable across tasks. Close Drawer ends in an initialization fall. The visualizations show recorded robot states without task objects; they do not depict autonomous policy evaluation.