ARIA: ASSISTIVE TELEOPERATION FOR HUMANOID LOCO-MANIPULATION VIA RESIDUAL INTERACTION REFINEMENT

Anonymous authors
Paper under double-blind review



Figure 1: ARIA is a co-pilot for humanoid teleoperation: the operator drives the whole-body motion, while ARIA refines the interaction with the object, enabling real-world box pickup and carrying.

ABSTRACT

Whole-body teleoperation enables an operator to drive a humanoid as a physical avatar, but this intuitive body-level mapping has a blind spot: it cannot close the loop at the contact interface with objects. Kinematic mismatches between human and robot cause grasps to fail and objects to slip, yet the operator — limited by a restricted view of the contact and the cognitive load of coordinating locomotion with manipulation — cannot compensate in real time. Therefore, we introduce ARIA (Assistive Residual Interaction for Avatars), a co-pilot that closes this loop on the operator's behalf: it augments a frozen teleoperation policy with bounded, perception-gated corrections to task-relevant joints conditioned on egocentric object geometry, and reverts to pure teleoperation when perception is unavailable. In simulation, on three loco-manipulation tasks (picking up a box from a platform or from the ground, and carrying a table), ARIA achieves an average success rate of 84.2%, where all whole-body teleoperation baselines fail, and reduces the object error by 36–59% relative to the best baseline while keeping the root-relative tracking error within the range of the baselines. Deployed on a Unitree G1 with PICO VR or a motion-capture suit and ArUco-based onboard perception, ARIA enables assisted box pickup where unassisted teleoperation fails.

1 INTRODUCTION

Humanoid robots share the morphological structure of the human body, making them natural physical avatars that extend human reach through whole-body teleoperation (Darvish et al., 2023). An operator can control the robot by simply moving their own body, bypassing joysticks and scripted commands for intuitive control (He et al., 2024b;a); with a wearable motion-capture suit (Zuo et al., 2025), not even a headset or handheld controllers are needed. Recent learning-based methods have rapidly advanced this paradigm, enabling scalable whole-body imitation (Ze et al., 2025), long-horizon closed-loop tracking (Li et al., 2025; Zhu et al., 2026), low-latency extremity control (Xiong et al., 2026), and expert-gated motion priors (Li et al., 2026).

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.

Current systems can faithfully track gross body motion such as walking, reaching, and gesturing, but this fidelity does not extend to robot-object contact. Kinematic mismatches between human and robot compromise object interaction, leading to inaccurate contacts and manipulation failures (Yang et al., 2025; Zhao et al., 2025a). The operator, further burdened by a restricted view of the contact, robot self-occlusion, and the cognitive load of coordinating locomotion with manipulation, cannot reliably compensate in real time. Existing teleoperated loco-manipulation systems (Stilman et al., 2008; Penco et al., 2019; Seo et al., 2023; Lu et al., 2024; Purushottam et al., 2024; Ben et al., 2025) advance control interfaces and stability, but largely leave object interaction to the operator.

This challenge lies at the intersection of two related lines of work, each providing a partial solution. **Assistive teleoperation** combines operator commands with robot autonomy to reduce operator burden through techniques such as policy blending (Dragan & Srinivasa, 2013), residual corrections (Schaff & Walter, 2020), latent actions (Jeon et al., 2020), human-in-the-loop learning (Chen et al., 2022), generative shared autonomy (Yoneda et al., 2023; Sun et al., 2025), vision-language intent inference (Liu et al., 2025), and sim-to-real copilots (Sha et al., 2026). On humanoid platforms, this paradigm has been applied to haptic feedback (Abi-Farraj et al., 2018), mixed-reality task specification (Penco et al., 2024), and support-contact completion (Rouxel et al., 2024); however, these methods target operator awareness, command generation, or balance feasibility, leaving object interaction unaddressed. **Object-aware autonomous methods** from imitation learning (Xu et al., 2025; Yang et al., 2025; Zhao et al., 2025a; Weng et al., 2025; Yin et al., 2025) show that reliable manipulation requires explicit modeling of body-object geometry and contact dynamics, yet they operate without a human in the loop, and their corrections are neither bounded to respect operator commands nor gated for safe real-time composition.

We present **ARIA** (Assistive Residual Interaction for Avatars), an **assistive teleoperation** framework for humanoid loco-manipulation through *residual interaction refinement* (Fig. 1). The key idea is a learned co-pilot: a residual policy that applies small, bounded corrections to task-relevant joints, conditioned on egocentric object perception. The operator retains control over locomotion and gross motion through a frozen base teleoperation controller, with the co-pilot automatically modulating its corrections based on real-time object observation. In simulation, ARIA succeeds in 100.0%, 52.5%, and 100.0% of the test episodes on Platform Pickup, Ground Pickup, and Table Carry, respectively, whereas all whole-body teleoperation baselines fail on every task.

Our contributions are as follows:

- We identify **object-interaction assistance** as a distinct, underexplored problem in humanoid teleoperation, arguing that intuitive object interaction through teleoperation requires learned assistive autonomy.
- We propose a **perception-gated residual policy** that applies bounded corrections to task-relevant joints based on egocentric object geometry, composing with a frozen base teleoperation controller.
- We realize the proposed framework on a Unitree G1 humanoid, teleoperated through either PICO VR or a wearable motion-capture suit, and validate its effectiveness. Simulation experiments on three loco-manipulation tasks show that ARIA succeeds where all whole-body teleoperation baselines fail and reduces the object error by 36–59%, and ARIA is successfully deployed on the real robot with both interfaces.

2 Related Work

2.1 Teleoperated Humanoid Loco-Manipulation

Humanoid teleoperation has progressed from joystick and multimode interfaces (Stilman et al., 2008; Penco et al., 2019) to learned whole-body control. Motion-tracking controllers now reproduce operator motion with increasing fidelity and generality (He et al., 2024b;a; Fu et al., 2024; Ze et al., 2025; Chen et al., 2026; Wu et al., 2026a; Luo et al., 2025), yet they are optimized for tracking rather than for the outcome of object contact. Teleoperated loco-manipulation systems pair such control with VR interfaces (Seo et al., 2023), motion priors (Lu et al., 2024), force feedback (Purushottam et al., 2024), and exoskeleton cockpits (Ben et al., 2025). Recent work adds robot-side adaptation: Baek et al. (2025) estimate object inertia online to adapt the robot's equilib-

rium and haptic feedback, and, outside teleoperation, FALCON learns force-adaptive policies for payload transport and door opening (Zhang et al., 2026). Closest to our composition, gated residual coordination keeps body and hand controllers frozen and applies bounded, gated corrections to their outputs (Wu et al., 2026b), but it targets body–hand geometric consistency from motion references and proprioception, without observing the object or the interaction outcome. Across these systems, the operator's command is tracked, stabilized, or compensated, but not corrected according to the observed hand–object geometry.

2.2 Assistive Teleoperation and Residual Policy Learning

Assistive teleoperation and shared autonomy combine human commands with robot autonomy while preserving the operator's authority over task intent. Foundational formulations include policy blending with intent arbitration (Dragan & Srinivasa, 2013), latent actions (Jeon et al., 2020), and human-in-the-loop interface learning (Chen et al., 2022). Residual policy learning adds a learned correction to a fixed controller (Silver et al., 2018; Johannink et al., 2019); Schaff & Walter (2020) brought it to shared autonomy as bounded corrections to user commands. Later work extends assistance with generative models (Yoneda et al., 2023; Sun et al., 2025), VLM-based intent inference (Liu et al., 2025), impedance copilots (Wang et al., 2026), and sim-to-real copilots (Sha et al., 2026), but does not address whole-body balance or locomotion–manipulation coupling.

On humanoids, haptic feedback conveys balance state to the operator (Abi-Farraj et al., 2018), mixed-reality assistance completes manipulation commands from object detection and affordance templates (Penco et al., 2024), and Flow-MultiSupport completes support contacts for whole-body feasibility (Rouxel et al., 2024). These methods improve awareness, command specification, or feasibility, rather than correcting the operator's whole-body command at every step toward a successful object interaction.

2.3 Object-Aware Interaction

Object-aware methods show that reliable object interaction requires more than body-motion tracking, modeling body–object relations in motion synthesis (Li et al., 2023; 2024), physics-based imitation (Wang et al., 2023; Xu et al., 2025), and interaction skill learning (Wang et al., 2025; Gao et al., 2024; Allshire et al., 2025). For humanoid loco-manipulation, OmniRetarget preserves robot–object–terrain contact relations during retargeting (Yang et al., 2025), HDMI learns humanoid–object interaction skills from monocular human videos via object co-tracking and interaction rewards (Weng et al., 2025), and VisualMimic couples egocentric vision with hierarchical whole-body control (Yin et al., 2025). Most related, ResMimic refines a general motion tracker with a residual policy trained on object-tracking and contact rewards (Zhao et al., 2025a), and AdaptManip pairs a whole-body base policy with residual manipulation control and recurrent object-state estimation for object lifting and delivery (Byrd et al., 2026). These methods execute tasks autonomously from reference motions or task goals. We instead bring object-aware residual correction into the teleoperation loop: corrections are computed online on top of the operator's command, bounded and gated so that the command remains the nominal behavior.

3 Method

3.1 Assistive Teleoperation Framework

Whole-body teleoperation policies reproduce the operator's motion faithfully but are unaware of the manipulated object, so small kinematic discrepancies between human and robot accumulate at the contact interface. We address this with an assistive teleoperation framework that augments the operator's commands with a learned, object-aware correction (Fig. 2). The framework consists of three components: a frozen base teleoperation policy that tracks the operator's motion, a residual assistive policy that corrects task-relevant joints conditioned on object perception, and a perception gate that determines when the correction is applied.

Residual Formulation. We formulate assistance as residual policy learning (Silver et al., 2018; Johannink et al., 2019). The joint position targets $a_t \in \mathbb{R}^{29}$ tracked by the PD controllers are

$$a_t = a_t^{\text{base}} + \Delta a_t^{\text{res}}, \tag{1}$$

where a_t^{base} is produced by the frozen base policy π_{base} from the operator's command, and Δa_t^{res} is a bounded correction produced by the assistive policy π_{res} and modulated by a perception gate $\eta_t \in [0, 1]$ (Eqs. 4 and 7). This decomposition keeps the operator's command as the nominal behavior and confines learning to the deviation from it.

Base Teleoperation Policy. The base policy π_{base} is a pretrained whole-body motion tracking controller that maps the operator's motion to joint targets at 50 Hz:

$$a_t^{\text{base}} = \pi_{\text{base}}\big(g_t^h,\, [\, s_t^p,\, a_{t-1}^{\text{base}}\,]_{t-9:t}\big), \qquad s_t^p = \big[\, q_t,\ \dot{q}_t,\ \omega_t,\ \gamma_t \,\big], \tag{2}$$

where g_t^h is the operator's SMPL body motion command, $q_t, \dot{q}_t \in \mathbb{R}^{29}$ are the joint positions and velocities, ω_t is the root angular velocity, and γ_t is the gravity direction in the root frame. It provides balance, locomotion, and general motion tracking, and is kept frozen throughout training and deployment. Because π_{base} conditions on its own previous outputs a_{t-1}^{base} rather than on the executed actions, the residual does not enter its action history.

Task-Relevant Residual. The base policy already provides reliable balance and locomotion, whereas the failures of unassisted teleoperation concentrate at the hand–object interface. We therefore restrict the residual to the task-relevant joints $\mathcal{J}_a$ that directly shape the hand–object interaction, namely the 14 arm joints (Appendix A.1):

$$\Delta a_t^{\text{res}}[i] = 0, \quad \forall\, i \notin \mathcal{J}_a. \tag{3}$$

Joints outside $\mathcal{J}_a$, which primarily govern balance and locomotion, receive no correction and follow π_{base} alone, so the correction does not act directly on whole-body stability.

Perception-Gated Activation. Assistance is useful only near the object and trustworthy only when the object is reliably perceived. Analogous to the arbitration function in shared-control policy blending (Dragan & Srinivasa, 2013), we modulate the residual with a gate that depends on hand–object proximity and perception validity:

$$\eta_t = v_t \cdot \text{clip}\left(\frac{d_{\text{off}} - d_t}{d_{\text{off}} - d_{\text{on}}},\, 0,\, 1\right), \tag{4}$$

where $d_t = \min(d_t^R, d_t^L)$ is the smaller of the two hand-to-object surface distances (Eq. 6), $d_{\text{on}} < d_{\text{off}}$ are the distances at which assistance becomes fully active and fully inactive (Appendix A.1), and $v_t \in \{0, 1\}$ indicates whether a valid object pose estimate is available. The gate ramps the correction in as the hands approach the object, avoiding abrupt changes in the commanded action. Whenever perception fails, $v_t{=}0$ and the system reverts to unassisted teleoperation, so the operator always retains control through π_{base}.

3.2 Residual Assistive Policy

The residual policy π_{res} maps an observation o_t to a raw action $\pi_{\text{res}}(o_t)$, which is converted to the bounded correction Δa_t^{res} by Eq. 7. The observation combines the robot state, the base action, and an egocentric representation of the object:

$$o_t = \big[\, [\, s_t^p,\ a_{t-1},\ a_t^{\text{base}},\ \phi_t^h\,]_{t-10:t},\ \ s_t^o \,\big]. \tag{5}$$

3.2.1 Observation Space

Robot state and base action. The residual policy observes the same proprioceptive state s_t^p as the base policy, together with the previously executed action a_{t-1}, which, unlike a_{t-1}^{base}, includes its own correction. Instead of the full motion command g_t^h, it observes the base action a_t^{base}, which already expresses the operator's intent at the joint level, together with the root roll, pitch, and yaw rate ϕ_t^h of the command. A causal history over the past 10 frames, in addition to the current frame, provides temporal context for contact dynamics.

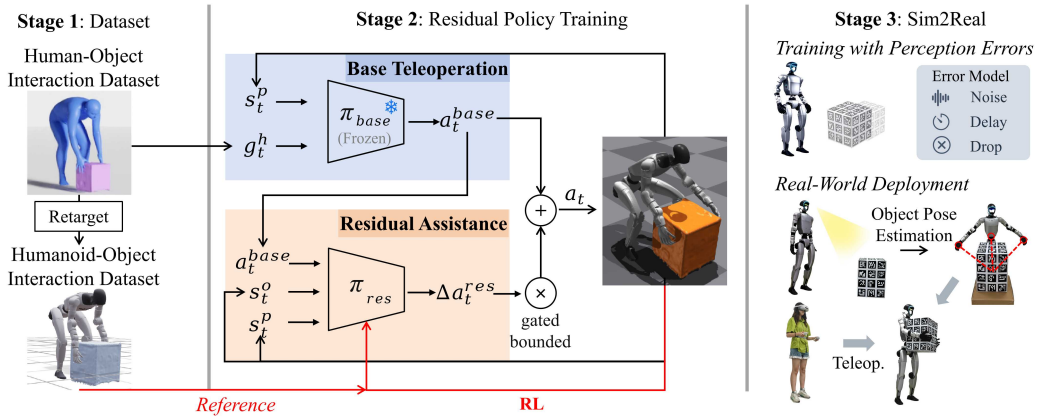

Figure 2: Overview of ARIA. Left: human–object demonstrations are retargeted to the robot and serve as training references. Middle: a frozen base teleoperation policy maps the operator's motion to joint targets, and a residual assistance policy trained with RL adds bounded, perception-gated corrections conditioned on the object state. Right: sim-to-real transfer, with training under a perception error model and real-world deployment with marker-based object perception and whole-body teleoperation.

Egocentric object geometry. We represent the object state as a compact 15-dimensional vector expressed in body-centric frames:

$$s_t^o = \left[\, p_t^{o|P},\ \theta_t^{o|P},\ p_t^{o|R},\ p_t^{o|L},\ d_t^R,\ d_t^L \,\right], \tag{6}$$

where $p_t^{o|P} \in \mathbb{R}^3$ and $\theta_t^{o|P} \in \mathbb{R}^4$ are the object position and orientation (as a quaternion) in the pelvis frame P, $p_t^{o|R}, p_t^{o|L} \in \mathbb{R}^3$ are the object position in the right and left hand frames, and $d_t^R, d_t^L \in \mathbb{R}_{\geq 0}$ are the hand-to-object surface distances. The surface distance $d_t^h = \min_j \|p_t^{h|o} - S[j]\|_2$, $h \in \{R, L\}$, is measured between the hand position $p_t^{h|o}$ and a dense set of points $S \in \mathbb{R}^{M\times 3}$ sampled on the object surface, both expressed in the object frame. This representation is invariant to the robot's global position and directly encodes the reach and alignment relevant to grasping. The surface distances additionally drive the perception gate (Eq. 4).

3.2.2 Bounded Residual Output

To keep the correction within a safe envelope around the operator's command, the raw policy output is gated, scaled, and smoothly saturated:

$$\Delta a_t^{\text{res}}[i] = c_i \tanh\left(\frac{\eta_t\, \beta_i\, \pi_{\text{res}}(o_t)[i]}{c_i}\right), \quad i \in \mathcal{J}_a, \tag{7}$$

where β_i is a per-joint gain and c_i a per-joint cap in radians (Appendix A.1). The $\tanh$ saturation guarantees $|\Delta a_t^{\text{res}}[i]| \leq c_i$ for any network output and gate value. Together with the joint mask (Eq. 3), this bounds the deviation from the base action at all times.

3.2.3 Training

Throughout training, π_{base} remains frozen and only π_{res} is optimized.

Reference motions. We train on human–object interaction data retargeted to the G1 with OmniRetarget (Yang et al., 2025), which preserves the hand–object spatial relationship, and augment it to increase diversity (Appendix B.2); the training and test sets are split by subject (Appendix B.4). At each step, the frozen π_{base} takes the human motion of the clip as g_t^h to produce a_t^{base}, and the retargeted robot and object trajectories serve as tracking references for π_{res}.

Reward. The reward combines object, contact, and motion terms (Appendix A.3). The main object term measures the alignment of object surface points with their reference (Zhao et al., 2025a),

capturing translational and rotational errors in a single term:

$$r_t^o = \left(1 + \lambda_o \frac{1}{N} \sum_{i=1}^{N} \|\boldsymbol{P}[i]_t - \hat{\boldsymbol{P}}[i]_t\|_2\right)^{-1}, \tag{8}$$

where $\boldsymbol{P}_t, \hat{\boldsymbol{P}}_t \in \mathbb{R}^{N\times 3}$ are the current and reference positions of the N surface points. The remaining object and contact terms reward matching the reference hand-to-object distances and hand placement during grasping and carrying. Since π_{base} already tracks the operator's motion, the motion term only penalizes upper-body deviations from the reference beyond a tolerance, and the residual is regularized by its magnitude and temporal change.

Optimization. We optimize π_{res} with PPO (Schulman et al., 2017) at 50 Hz. The joint mask (Eq. 3) is also applied to the policy-gradient objective, so that only the task-relevant action dimensions receive gradients. The final actor layer uses small-gain Xavier initialization (Ankile et al., 2024), so that training starts from near-zero residuals, i.e., from the behavior of π_{base}. The network architecture and the PPO hyperparameters are given in Appendices A.2 and A.4.

3.3 Real-World Deployment

We deploy the framework on a Unitree G1 humanoid. In simulation, the object pose is provided by the physics engine; on hardware, it must be estimated from onboard sensing. We describe the object pose estimation pipeline, the perception error model that bridges this gap during training, and the teleoperation interfaces.

3.3.1 Marker-Based Object Pose Estimation

We attach ArUco markers to multiple faces of the box, each associated with a known pose in the object frame. The robot's head-mounted camera detects the visible markers and, whenever at least four markers of one face are visible, estimates the object pose $T_t^{o|C}$ in the camera frame C via PnP; the multi-face layout provides redundancy under partial occlusion. The camera-to-pelvis kinematic chain, given by forward kinematics and the nominal camera mounting (Appendix E.2), then yields the pelvis-frame pose $T_t^{o|P}$, from which s_t^o is computed as in simulation.

3.3.2 Perception Error Model

Marker-based perception introduces noise, latency, and intermittent failures that are absent in simulation. To make the deployed π_{res} robust to these imperfections, we train it with a stochastic perception model in place of the ground-truth object pose, reproducing the characteristics of the real pipeline: (i) Gaussian noise on the camera-frame position (σ_p=5 mm) and orientation (σ_θ=1°); (ii) a transport delay of 20–80 ms per measurement, modeling detection and communication latency; (iii) random frame dropout (5%) and burst occlusions of 100–300 ms (1% onset probability); (iv) a per-environment calibration bias (σ=2 mm / 0.3°), modeling camera extrinsic error; and (v) a geometric visibility model that determines the visible markers from the camera field of view, the viewing angle, and self-occlusion by the robot's arms. All perturbations are applied to the camera-frame pose, from which s_t^o is then computed. If the latest valid measurement is older than 100 ms, the estimate is treated as invalid (v_t=0): s_t^o is zeroed and the gate closes. This model serves sim-to-real transfer only: the simulation experiments (Sec. 4.2) train and evaluate π_{res} with the ground-truth object pose, so that they measure the assistance itself rather than the perception quality.

3.3.3 Teleoperation Interfaces

The operator wears a PICO VR headset with body tracking, which estimates the operator's SMPL body pose in real time and streams it to π_{base} as g_t^h. Toward more natural, avatar-like teleoperation, the operator can instead wear an inertial motion-capture suit (Zuo et al., 2025), which provides the SMPL body pose without a headset or handheld controllers, so that the operator drives the robot simply by moving their own body. Since π_{res} observes a_t^{base} rather than g_t^h, both interfaces use the same policies without modification. With either interface, the assistive layer is transparent to the operator: they teleoperate the robot as usual, and the residual improves object interaction without requiring them to adapt to the assistance.

Table 1: Simulation results. SR is the success rate (%); E_{obj} is the object error, and E_{G} / E_{L} are the global / local MPJPE, all in centimeters. The baselines track the demonstrated motion without task-aware correction; ARIA adds a learned residual.

	Platform Pickup				Ground Pickup				Table Carry			
Method	SR↑	E_{obj}↓	E_{G}↓	E_{L}↓	SR↑	E_{obj}↓	E_{G}↓	E_{L}↓	SR↑	E_{obj}↓	E_{G}↓	E_{L}↓
SONIC	0.0	70.8	20.9	4.9	0.0	123.9	**22.3**	7.2	0.0	81.3	23.6	8.9
HoloMotion	0.0	100.7	28.0	5.5	0.0	106.3	23.2	5.2	0.0	83.0	**10.7**	3.4
TeleopIt	0.0	85.9	**14.9**	**2.5**	0.0	113.2	22.6	**4.1**	0.0	81.3	14.9	**3.2**
ARIA	**100.0**	**28.8**	26.7	5.4	**52.5**	**68.0**	29.5	6.5	**100.0**	**42.6**	38.3	8.2

4 EXPERIMENTS

4.1 EXPERIMENT SETUP

Tasks. We evaluate ARIA on three loco-manipulation tasks that differ in the interaction object and in how much whole-body coordination they require. **(1) Platform Pickup:** the robot lifts a box off an elevated platform; the box is within reach of a mild forward bend, so the balance requirement is relatively low. **(2) Ground Pickup:** the robot picks up and carries a box from the ground, requiring deep bending or squatting and tight coordination among balance, locomotion, and object interaction. **(3) Table Carry:** the robot lifts and carries a table from the ground by gripping the edges of its tabletop; compared with the box, the table has a different shape and grasp geometry. In all tasks, each test clip plays the role of the operator: its recorded motion is replayed open-loop as the command, without reacting to the robot. The test sets contain 100, 80, and 100 clips from held-out subjects (Appendix B.4).

Metrics. We report three metrics. **(1) Success Rate (SR, %)** counts an episode as successful when three conditions hold: the robot does not fall, the object is lifted (its lowest point rises 0.05 m above its initial height for Platform Pickup and 0.20 m for the other tasks), and the lift is maintained for at least half of the reference hold duration (Appendix D.1). **(2) Object Error (E_{obj}, cm)** is the mean distance between the simulated and reference positions of the object surface points (Eq. 8), averaged over all frames. **(3) MPJPE (E_{G} / E_{L}, cm)** measures humanoid motion tracking as the mean per-joint position error with respect to the reference robot motion, computed in the world frame (E_{G}) and relative to the root (E_{L}).

Comparison Methods. We compare against three whole-body teleoperation methods: **SONIC** (Luo et al., 2025), **HoloMotion** (Chen et al., 2026), and **TeleopIt** (Wu et al., 2026a). SONIC serves as both the frozen base policy in our framework and a standalone baseline. None of the baselines applies a task-aware correction: SONIC tracks the human motion, as the base policy does in our framework, whereas HoloMotion and TeleopIt track the retargeted robot motion (Appendix C). **ARIA** (ours) augments the output of the frozen SONIC with a learned residual policy that corrects task-relevant joints conditioned on object perception, which in simulation is the ground-truth object pose.

4.2 SIMULATION EVALUATION

Table 1 and Fig. 3 report the simulation results; Appendix D.2 breaks the success rate down into its three conditions. No baseline completes any of the three tasks, although all of them track the reference motion with a root-relative error of at most 8.9 cm. ARIA succeeds in 100.0%, 52.5%, and 100.0% of the test episodes on Platform Pickup, Ground Pickup, and Table Carry, and reduces the object error by 59%, 36%, and 48% relative to the best baseline (28.8 vs. 70.8, 68.0 vs. 106.3, and 42.6 vs. 81.3 cm). Its E_{L} stays within the range of the baselines on all three tasks, whereas E_{G} is higher on Ground Pickup and Table Carry (29.5 and 38.3 cm vs. at most 23.2 and 23.6 cm). Motion tracking alone is thus insufficient for object interaction, and the residual closes this gap across tasks with different whole-body demands and object geometries.

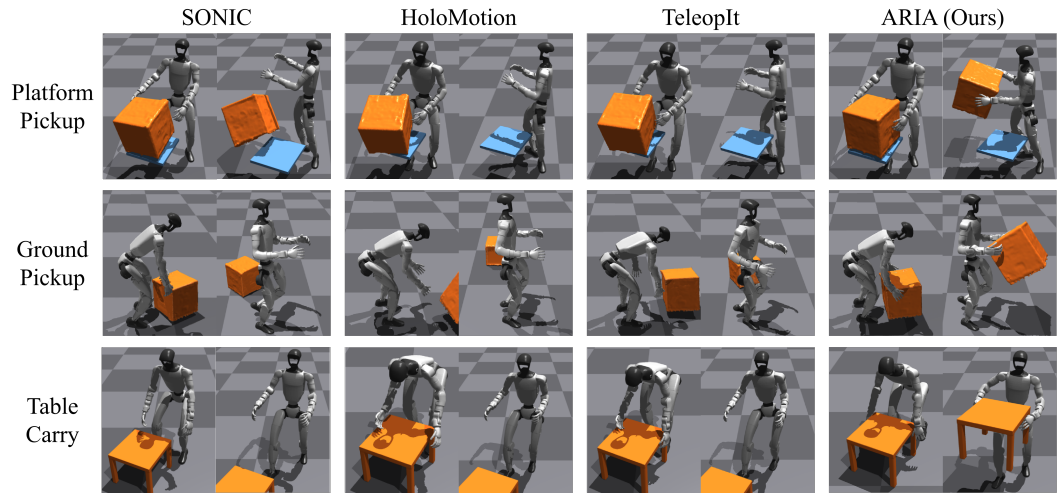

Figure 3: Qualitative comparison on the three tasks (rows) across methods (columns). The baselines track the reference motion but fail to grasp and lift the object, whereas ARIA applies object-aware residual corrections that enable successful grasping and lifting.

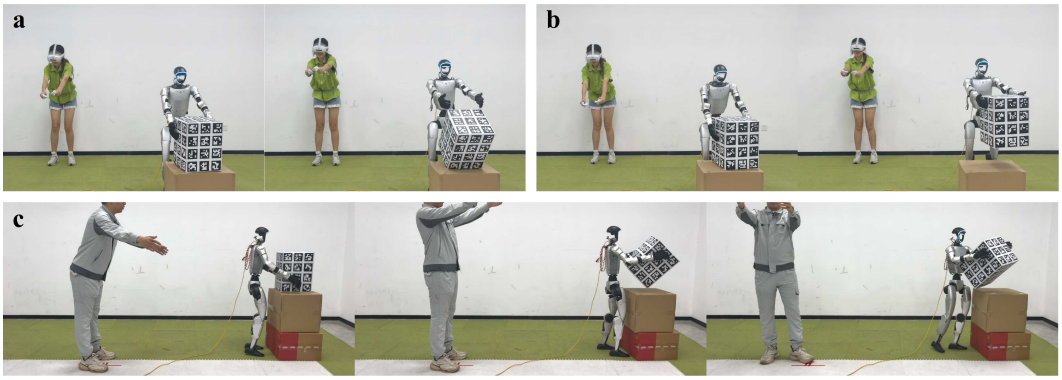

Figure 4: Real-world deployment on a Unitree G1. (a) SONIC with PICO VR: the robot follows the operator, but the box slips out of its hands. (b) ARIA with PICO VR: the correction establishes a stable grasp, and the robot lifts the box. (c) ARIA with a motion-capture suit: the operator drives the robot with their own body, and the robot lifts the box.

4.3 Real-World Demonstration

We deploy ARIA on a Unitree G1 humanoid with the teleoperation interfaces and marker-based object perception described in Sec. 3.3, with both policies running at 50 Hz on an off-board workstation (Appendix E). With PICO VR and the same operator (Fig. 4(a) and (b)), SONIC alone follows the operator's motion but lets the box slip out of its hands, whereas ARIA's gated correction takes effect as the hands approach the perceived box and establishes a stable grasp, so that the robot lifts the box while the operator retains control over the whole-body motion. With the motion-capture suit (Fig. 4(c)), the operator teleoperates without a headset or handheld controllers, and the same policies again enable the robot to grasp and lift the box, so ARIA transfers to the real robot with both interfaces.

Table 2: Ablation results. Metrics are the same as in Table 1.

	Platform Pickup				Ground Pickup				Table Carry			
Method	SR↑	$E_{\text{obj}}\downarrow$	$E_{\text{G}}\downarrow$	$E_{\text{L}}\downarrow$	SR↑	$E_{\text{obj}}\downarrow$	$E_{\text{G}}\downarrow$	$E_{\text{L}}\downarrow$	SR↑	$E_{\text{obj}}\downarrow$	$E_{\text{G}}\downarrow$	$E_{\text{L}}\downarrow$
Fine-tuned SONIC	64.0	41.3	30.8	11.4	66.3	63.2	33.6	8.9	50.0	48.1	36.2	10.8
w/o geometry	50.0	83.5	27.2	5.2	25.0	91.8	30.8	6.1	27.0	75.9	24.0	8.2
w/o gate	85.0	45.2	27.9	5.6	42.5	77.7	41.0	7.6	51.0	107.0	110.1	16.7
ARIA	100.0	28.8	26.7	5.4	52.5	68.0	29.5	6.5	100.0	42.6	38.3	8.2

4.4 Ablation Study

We ablate three design choices of ARIA (Table 2; setup details in Appendix C). **Fine-tuned SONIC** replaces the residual policy by fine-tuning the base policy itself on the same data and rewards. **w/o geometry** removes the object geometry s_t^o from the residual observation (Eq. 5), while the gate η_t still uses the hand-to-object distances. **w/o gate** sets $\eta_t \equiv 1$ at test time, so that the trained residual is applied regardless of the hand–object distance.

Fine-tuned SONIC. Fine-tuning the base policy surpasses ARIA only on Ground Pickup (66.3% vs. 52.5%), falls well below it on Platform Pickup and Table Carry (64.0% and 50.0% vs. 100.0%), and has a higher E_{L} than both SONIC and ARIA on all three tasks. The residual design therefore yields more consistent success while better preserving the operator's motion.

Object geometry. Without object geometry, the success rate drops on all three tasks (50.0%, 25.0%, and 27.0% vs. 100.0%, 52.5%, and 100.0%), and the object error increases (83.5, 91.8, and 75.9 vs. 28.8, 68.0, and 42.6 cm), while E_{L} stays within 0.4 cm of ARIA, so the drop does not stem from degraded tracking. Since the gate still restricts the correction to the vicinity of the object, knowing when to assist is not sufficient; the correction relies on the perceived object state to decide how to assist.

Perception gate. Without the gate, the success rate drops on all three tasks (85.0%, 42.5%, and 51.0% vs. 100.0%, 52.5%, and 100.0%), and both E_{obj} and E_{G} increase. Applying the correction while the hands are still far from the object disturbs the operator's motion and hinders the subsequent grasp.

5 Conclusion and Limitations

We presented ARIA, an assistive teleoperation framework that composes a frozen base teleoperation policy with a perception-gated, bounded residual co-pilot on task-relevant joints, conditioned on egocentric object geometry. It bridges the gap between the operator's tracked motion and successful object interaction without modifying the base controller, and reverts to unassisted teleoperation when object sensing is unavailable.

Experiments show that existing whole-body teleoperation methods fail on all three tasks despite reasonable motion tracking, whereas ARIA succeeds on 52.5–100.0% of test episodes while keeping its root-relative tracking error within the range of the baselines. Real-world deployment on a Unitree G1 with ArUco-based perception validates the sim-to-real pipeline with both PICO VR and a motion-capture suit, the latter pointing toward more natural, avatar-like teleoperation. Ablations show that both the perception gate and the object geometry contribute to success on every task, and that residual assistance preserves tracking quality better than fine-tuning the base policy, which surpasses ARIA in success rate only on Ground Pickup.

Limitations. Our evaluation covers two rigid objects, a box and a table; generalization to more diverse geometries and to deformable objects requires further investigation. The perception system relies on ArUco markers affixed to the object, limiting applicability to prepared targets; replacing it with learned, marker-free pose estimation would broaden deployment to unstructured environments. Finally, the robot's built-in simplified hands preclude dexterous finger control; integrating multi-fingered hands would enable finer grasp strategies.

AI USE STATEMENT

We used generative AI tools to polish the writing, to find related work, and to implement code. We did not use them to develop research ideas, design the method, or generate data. All AI-assisted text and references were checked by the authors, and all AI-assisted code was verified and tested. We take full responsibility for the content of this work.

ETHICS STATEMENT

We use the publicly released OMOMO dataset (Li et al., 2023) in accordance with its license and terms of use; it contains body-motion parameters and object poses without personally identifying information. We did not collect any new human-subject data. Real-robot experiments were operated by the authors, and the operator's tracked pose was used only for real-time control and was not stored. The assistive correction is bounded and rate-limited, falls back to pure teleoperation when perception is unavailable, and a safety tether or a safety spotter was used throughout all hardware trials. We are not aware of other ethical concerns raised by this work.

REPRODUCIBILITY STATEMENT

The method is described in Sec. 3.1–3.3, and the appendix provides the details needed to reproduce our results. Appendix A specifies the residual policy, network architecture, reward terms, and training hyperparameters. Appendix B describes the data sources, retargeting, data augmentation, task objects, and train/test splits. Appendix C details how each baseline is configured and evaluated and how the ablations are trained, and Appendix D gives the exact success criteria. Appendix E describes the real-robot hardware, marker calibration, runtime pipeline, and safety mechanisms.

REFERENCES

Firas Abi-Farraj, Bernd Henze, Alexander Werner, Michael Panzirsch, Christian Ott, and Maximo A. Roa Garzon. Humanoid teleoperation using task-relevant haptic feedback. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5010–5017, 2018.

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 *Conference on Robot Learning*, 2025.

Lars Ankile, Anthony Simeonov, Idan Shenfeld, and Pulkit Agrawal. Imitation refinement in residual policy learning. In *Conference on Robot Learning*, 2024.

Donghoon Baek, Amartya Purushottam, Jason J. Choi, and Joao Ramos. Whole-body bilateral teleoperation with multi-stage object parameter estimation for wheeled humanoid locomanipulation. *arXiv preprint arXiv:2508.09846*, 2025. URL https://arxiv.org/abs/2508.09846.

Qingwei Ben, Feiyu Jia, Jia Zeng, Junting Dong, Dahua Lin, and Jiangmiao Pang. Homie: Humanoid loco-manipulation with isomorphic exoskeleton cockpit. In *Robotics: Science and Systems (RSS)*, 2025. URL https://arxiv.org/abs/2502.13013.

Morgan Byrd, Donghoon Baek, Kartik Garg, Hyunyoung Jung, Daesol Cho, Maks Sorokin, Robert Wright, and Sehoon Ha. AdaptManip: Learning adaptive whole-body object lifting and delivery with online recurrent state estimation. *arXiv preprint arXiv:2602.14363*, 2026. URL https://arxiv.org/abs/2602.14363.

Maiyue Chen, Kaihui Wang, Bo Zhang, Xihan Ma, Zhiyuan Yang, Yi Ren, Qijun Huang, Zihao Zhu, Yucheng Wang, and Zhizhong Su. Holomotion-1 technical report. *arXiv preprint arXiv:2605.15336*, 2026. URL https://arxiv.org/abs/2605.15336.

Sean Chen, Jensen Gao, Siddharth Reddy, Glen Berseth, Anca D. Dragan, and Sergey Levine. Asha: Assistive teleoperation via human-in-the-loop reinforcement learning. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2022.

Kourosh Darvish, Luigi Penco, João Ramos, Rafael Cisneros, Jerry Pratt, Eiichi Yoshida, Serena Ivaldi, and Daniele Pucci. Teleoperation of humanoid robots: A survey. *IEEE Transactions on Robotics*, 39(3):1706–1727, 2023.

Anca D. Dragan and Siddhartha S. Srinivasa. A policy-blending formalism for shared control. *The International Journal of Robotics Research*, 32(7):790–805, 2013.

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

Jiawei Gao, Ziqin Wang, Zeqi Xiao, Jingbo Wang, Tai Wang, Jinkun Cao, Xiaolin Hu, Si Liu, Jifeng Dai, and Jiangmiao Pang. Coohoi: Learning cooperative human-object interaction with manipulated object dynamics. In *Advances in Neural Information Processing Systems*, 2024.

Sergio Garrido-Jurado, Rafael Muñoz-Salinas, Francisco José Madrid-Cuevas, and Manuel Jesús Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280–2292, 2014.

Tairan He, Zhengyi Luo, Xialin He, Wenli Xiao, Chong Zhang, Weinan Zhang, Kris Kitani, Changliu Liu, and Guanya Shi. Omnih2o: Universal and dexterous human-to-humanoid whole-body teleoperation and learning. In *Conference on Robot Learning (CoRL)*, 2024a.

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

Hong Jun Jeon, Dylan P. Losey, and Dorsa Sadigh. Shared autonomy with learned latent actions. In *Robotics: Science and Systems (RSS)*, 2020.

Tobias Johannink, Shikhar Bahl, Ashvin Nair, Jianlan Luo, Avinash Kumar, Matthias Loskyll, Juan Aparicio Ojea, Eugen Solowjow, and Sergey Levine. Residual reinforcement learning for robot control. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6023–6029, 2019.

Jiaman Li, Jiajun Wu, and C. Karen Liu. Object motion guided human motion synthesis. *ACM Transactions on Graphics*, 42(6), 2023.

Jiaman Li, Alexander Clegg, Roozbeh Mottaghi, Jiajun Wu, Xavier Puig, and C. Karen Liu. Controllable human-object interaction synthesis. In *European Conference on Computer Vision*, 2024.

Jie Li, Bing Tang, Feng Wu, and Rongyun Cao. Telegate: Whole-body humanoid teleoperation via gated expert selection with motion prior. *arXiv preprint arXiv:2602.09628*, 2026. URL https://arxiv.org/abs/2602.09628.

Yixuan Li, Yutang Lin, Jieming Cui, Tengyu Liu, Wei Liang, Yixin Zhu, and Siyuan Huang. Clone: Closed-loop whole-body humanoid teleoperation for long-horizon tasks. In *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, 2025. URL https://arxiv.org/abs/2506.08931.

Huihan Liu, Rutav Shah, Shuijing Liu, Jack Pittenger, Mingyo Seo, Yuchen Cui, Yonatan Bisk, Roberto Martín-Martín, and Yuke Zhu. Casper: Inferring diverse intents for assistive teleoperation with vision language models. In *Proceedings of the 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 2462–2483. PMLR, 2025.

Chenhao Lu, Xuxin Cheng, Jialong Li, Shiqi Yang, Mazeyu Ji, Chengjing Yuan, Ge Yang, Sha Yi, and Xiaolong Wang. Mobile-television: Predictive motion priors for humanoid whole-body control. *arXiv preprint arXiv:2412.07773*, 2024. URL https://arxiv.org/abs/2412.07773.

Zhengyi Luo, Ye Yuan, Tingwu Wang, Chenran Li, Fernando Castañeda, et al. Sonic: Supersizing motion tracking for natural humanoid whole-body control. *arXiv preprint arXiv:2511.07820*, 2025. URL https://arxiv.org/abs/2511.07820.

Luigi Penco, Nicola Scianca, Valerio Modugno, Leonardo Lanari, Giuseppe Oriolo, and Serena Ivaldi. A multimode teleoperation framework for humanoid loco-manipulation: An application for the icub robot. *IEEE Robotics & Automation Magazine*, 26(4):73–82, 2019.

Luigi Penco, Kazuhiko Momose, Stephen McCrory, Dexton Anderson, Nicholas Kitchel, Duncan Calvert, and Robert J. Griffin. Mixed reality teleoperation assistance for direct control of humanoids. *IEEE Robotics and Automation Letters*, 2024. arXiv:2411.01014.

Amartya Purushottam, Jack Yan, Christopher Xu, Youngwoo Sim, and Joao Ramos. Wheeled humanoid bilateral teleoperation with position-force control modes for dynamic loco-manipulation. *arXiv preprint arXiv:2407.12189*, 2024. URL https://arxiv.org/abs/2407.12189.

Quentin Rouxel, Andrea Ferrari, Serena Ivaldi, and Jean-Baptiste Mouret. Flow matching imitation learning for multi-support manipulation. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 528–535, 2024.

Charles Schaff and Matthew R. Walter. Residual policy learning for shared autonomy. In *Robotics: Science and Systems (RSS)*, 2020.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Mingyo Seo, Steve Han, Kyutae Sim, Seung Hyeon Bang, Carlos Gonzalez, Luis Sentis, and Yuke Zhu. Deep imitation learning for humanoid loco-manipulation through human teleoperation. *arXiv preprint arXiv:2309.01952*, 2023. URL https://arxiv.org/abs/2309.01952.

Shuo Sha, Yixuan Wang, Binghao Huang, Antonio Loquercio, and Yunzhu Li. Efficient and reliable teleoperation through real-to-sim-to-real shared autonomy. *arXiv preprint arXiv:2603.17016*, 2026.

Tom Silver, Kelsey Allen, Josh Tenenbaum, and Leslie Kaelbling. Residual policy learning. *arXiv preprint arXiv:1812.06298*, 2018.

Mike Stilman, Koichi Nishiwaki, and Satoshi Kagami. Humanoid teleoperation for whole body manipulation. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3175–3180, 2008.

Luzhe Sun, Jingtian Ji, Xiangshan Tan, and Matthew R. Walter. Flashback: Consistency model-accelerated shared autonomy. *arXiv preprint arXiv:2505.16892*, 2025.

Yeping Wang, Zhengtong Xu, Pornthep Preechayasomboon, Ben Abbatematteo, Amirhossein H. Memar, Nicholas Colonnese, and Sonny Chan. Stiffness copilot: An impedance policy for contact-rich teleoperation. *arXiv preprint arXiv:2603.14068*, 2026.

Yinhuai Wang, Jing Lin, Ailing Zeng, Zhengyi Luo, Jian Zhang, and Lei Zhang. Physhoi: Physics-based imitation of dynamic human-object interaction. *arXiv preprint arXiv:2312.04393*, 2023.

Yinhuai Wang, Qihan Zhao, Runyi Yu, Hok Wai Tsui, Ailing Zeng, Jing Lin, Zhengyi Luo, Jiwen Yu, Xiu Li, Qifeng Chen, Jian Zhang, Lei Zhang, and Ping Tan. Skillmimic: Learning basketball interaction skills from demonstrations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17540–17549, 2025.

Zhipeng Weng, Zhongang Cai, Jiaman Chen, Zhengyi Luo, C. Karen Liu, and Jiaman Li. Hdmi: Learning interactive humanoid whole-body control from human videos. *arXiv preprint arXiv:2509.16757*, 2025.

Bingqian Wu, Zicheng Xu, Xianghui Fan, Dayu Li, and Xiangru Huang. Teleopit: A full-embodiment humanoid teleoperation system. *arXiv preprint arXiv:2608.01834*, 2026a. URL https://arxiv.org/abs/2608.01834.

Ruiming Wu, Shuang Li, Liding Zhang, Alois Knoll, and Zhaopeng Chen. Gated residual body-hand coordination for whole-body humanoid teleoperation. *arXiv preprint arXiv:2609.18763*, 2026b. URL https://arxiv.org/abs/2609.18763.

Ziyan Xiong, Lixing Fang, Junyun Huang, Kashu Yamazaki, Hao Zhang, and Chuang Gan. Extremcontrol: Low-latency humanoid teleoperation with direct extremity control. *arXiv preprint arXiv:2602.11321*, 2026. URL https://arxiv.org/abs/2602.11321.

Sirui Xu, Hung Yu Ling, Yu-Xiong Wang, and Liang-Yan Gui. Intermimic: Towards universal whole-body control for physics-based human-object interactions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Yuxiang Yang, Zeyu Wang, Jinchao Huang, Zhengyi Luo, C. Karen Liu, and Guanya Shi. Omniretarget: Interaction-preserving data generation for humanoid whole-body loco-manipulation and scene interaction. *arXiv preprint arXiv:2509.26633*, 2025.

Shaofeng Yin, Yanjie Ze, Hong-Xing Yu, C. Karen Liu, and Jiajun Wu. Visualmimic: Visual humanoid loco-manipulation via motion tracking and generation. *arXiv preprint arXiv:2509.20322*, 2025.

Takuma Yoneda, Luzhe Sun, Ge Yang, Bradly C. Stadie, and Matthew R. Walter. To the noise and back: Diffusion for shared autonomy. In *Robotics: Science and Systems (RSS)*, 2023.

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. *arXiv preprint arXiv:2505.02833*, 2025. URL https://arxiv.org/abs/2505.02833.

Yuanhang Zhang, Yifu Yuan, Prajwal Gurunath, Ishita Gupta, Shayegan Omidshafiei, Ali-akbar Agha-mohammadi, Marcell Vazquez-Chanlatte, Liam Pedersen, Tairan He, and Guanya Shi. FALCON: Learning force-adaptive humanoid loco-manipulation. In *Learning for Dynamics and Control Conference (L4DC)*, 2026. URL https://arxiv.org/abs/2505.06776.

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. *arXiv preprint arXiv:2510.05070*, 2025a.

Zhigen Zhao, Liuchuan Yu, Ke Jing, and Ning Yang. XRoboToolkit: A cross-platform framework for robot teleoperation. *arXiv preprint arXiv:2508.00097*, 2025b. URL https://arxiv.org/abs/2508.00097.

Tengjie Zhu, Guanyu Cai, Zhaohui Yang, Guanzhu Ren, Haohui Xie, ZiRui Wang, Junsong Wu, Jingbo Wang, Xiaokang Yang, Yao Mu, and Yichao Yan. Clot: Closed-loop global motion tracking for whole-body humanoid teleoperation. *arXiv preprint arXiv:2602.15060*, 2026. URL https://arxiv.org/abs/2602.15060.

Chengxu Zuo, Jiawei Huang, Xiao Jiang, Yuan Yao, Xiangren Shi, Rui Cao, Xinyu Yi, Feng Xu, Shihui Guo, and Yipeng Qin. Transformer imu calibrator: Dynamic on-body imu calibration for inertial motion capture. *ACM Transactions on Graphics (TOG)*, 44(4):1–14, 2025.

A Implementation Details

A.1 Residual Policy

Task-relevant joints and bounds. The task-relevant joint set $\mathcal{J}_a$ contains the 14 arm joints, i.e., the shoulder pitch, roll, and yaw, the elbow, and the wrist roll, pitch, and yaw of each arm. The waist and leg joints receive no correction. Table 3 lists the gains β_i and caps c_i of Eq. 7. They are shared by both arms and all tasks; the small wrist cap avoids unnatural wrist twisting. For Platform Pickup, we additionally limit the rate of change of the correction to 1 rad/s per joint.

Perception gate. The gate distances in Eq. 4 are $(d_{\text{on}}, d_{\text{off}}) = (0.30, 0.60)$ m for Ground Pickup and Table Carry, so that the correction starts during the approach to the object. In Platform Pickup, the hands start close to the box; we therefore train with $(0.15, 0.35)$ m but tighten the gate to $(0.06, 0.10)$ m at test time, so that the correction is applied only once the hands reach the box.

Table 3: Gains β_i and caps c_i of the bounded residual (Eq. 7), shared by both arms and all tasks.

Joint	Gain β_i	Cap c_i (rad)
Shoulder pitch, roll, yaw	0.20	0.25
Elbow	0.45	0.375
Wrist roll, pitch, yaw	0.50	0.16

A.2 Network Architecture

The actor and the critic are MLPs with hidden sizes $(512, 512, 256, 128)$, SiLU activations, and layer normalization before the last hidden layer. At each frame, the actor observes s_t^p, a_{t-1}, and a_t^{base} (Eq. 5), together with the root roll, pitch, and yaw rate ϕ_t^h of the command. The 10 history frames are compressed by a temporal convolutional encoder, and the resulting latent is concatenated with the current frame and the object geometry s_t^o as the input of the MLP. The critic additionally receives privileged information, including the root linear velocity, key-body positions, foot contacts, and the randomized physical parameters. The final actor layer is initialized with Xavier gain 0.02, and the initial action standard deviation is 0.2.

A.3 Reward

Table 4 lists all reward terms. The object terms compare the $N{=}256$ surface points of the object and the hand-to-object surface distances d_t^h (Eq. 6) with their reference values, marked by $\hat{\cdot}$. The contact label $\hat{c}_t^h \in \{0, 1\}$ of hand $h \in \{R, L\}$ is set when the reference hand is within 0.25 m of the reference object surface, so the contact terms are active only in the grasping and carrying phases of the reference. Since π_{base} already tracks the operator's motion, the motion term does not reward joint tracking; it only penalizes upper-body joints that deviate from the reference beyond a per-group tolerance τ_g (0.42, 0.25, 0.16, and 0.30 rad for the shoulder, elbow, wrist, and waist). The residual itself is regularized by its magnitude and temporal change.

Table 4: Reward terms. $\hat{\cdot}$ denotes reference values, p_t^h is the position of hand h, and $\hat{c}_t = \max_h \hat{c}_t^h$. avg_h averages over the hands with $\hat{c}_t^h = 1$; e_t^{arm} is the squared error of the forearm and upper-arm vectors. The group weights w_g are 2, 4, 7, and 1 for the shoulder, elbow, wrist, and waist. *The upper-body deviation weight is -10 for Platform Pickup. The regularization terms and their weights follow Zhao et al. (2025a).

Term	Definition	Weight
Object		
Object point tracking	$1/\big(1 + 3\,\frac{1}{N}\sum_i \|\boldsymbol{P}[i]_t - \hat{\boldsymbol{P}}[i]_t\|_2\big)$	3.0
Hand–object distance	$1/\big(1 + 10\,\frac{1}{2}\sum_h \lvert d_t^h - \hat{d}_t^h\rvert\big)$	4.0
Contact		
Contact distance	$\text{avg}_h\, 1/\big(1 + 10\,\lvert d_t^h - \hat{d}_t^h\rvert\big)$	2.0
Contact hand position	$\text{avg}_h\, 1/\big(1 + 6\,\|p_t^h - \hat{p}_t^h\|_2\big)$	2.0
Arm shape	$\hat{c}_t/(1 + 18\, e_t^{\text{arm}})$	1.0
Bimanual grasp	hands on opposite sides of the object, centered and level	0.75
Approach	hand–object closing speed before contact, clipped at 0.5 m/s	0.01
Motion and regularization		
Upper-body deviation	$\sum_g w_g\, \text{mean}_{i\in g} \max(0, \lvert q_t[i] - \hat{q}_t[i]\rvert - \tau_g)^2$	-3^*
Residual magnitude	$\text{mean}_{i\in\mathcal{J}_a} \Delta a_t^{\text{res}}[i]^2$	-10^{-3}
Residual rate	$\text{mean}_{i\in\mathcal{J}_a} (\Delta a_t^{\text{res}}[i] - \Delta a_{t-1}^{\text{res}}[i])^2$	-5×10^{-3}
Regularization	action rate, joint velocity, acceleration, and limits, foot slip and contact forces	–

A.4 Training Hyperparameters

All residual policies are trained in Isaac Gym with 4096 parallel environments, a simulation step of 2 ms, and a control frequency of 50 Hz; Table 5 lists the PPO hyperparameters. Each episode starts from the first frame of a training clip, with the initial velocity set to the reference velocity scaled by a factor sampled from $\mathcal{U}[0.8, 1]$. An episode terminates at the end of the clip, when the pelvis height deviates from the reference by more than 0.3 m, or when the mean object surface point error exceeds 1.0 m. We randomize the ground friction ($\mathcal{U}[0.1, 2.0]$), the base mass (± 3 kg) and center of mass (± 5 cm), the motor strength ($\mathcal{U}[0.8, 1.2]$), and the gravity ($\pm 0.1\,\mathrm{m/s^2}$), and add uniform noise to the proprioceptive observations.

Table 5: PPO hyperparameters.

Hyperparameter	Value
Parallel environments	4096
Steps per environment	24
Learning epochs	5
Mini-batches	4
Learning rate	2×10^{-4} (adaptive, target KL 0.008)
Discount factor	0.99
GAE λ	0.95
Clip ratio	0.2
Entropy coefficient	0.005
Value loss coefficient	1.0
Max gradient norm	1.0

B Data Processing and Splits

B.1 Data Sources and Retargeting

All reference motions are derived from OMOMO (Li et al., 2023), which provides human SMPL-X motions with synchronized 6-DoF poses of scanned objects. We use the sequences in which a subject lifts and carries the large box (Platform Pickup and Ground Pickup) or the large table (Table Carry) with both hands. For a subset of box motions, we first refine the human motion around pickup and placement, correcting implausible grasps and excessive trunk flexion with inverse kinematics, so that the subsequent robot motion inherits a feasible grasp. The human–object motions are then retargeted to the 29-DoF Unitree G1 with OmniRetarget (Yang et al., 2025), which preserves the hand–object spatial relationship. Each resulting clip contains the human motion, the robot motion, and the object trajectory at 30 Hz. Before training, we ground each clip so that the object rests on its support surface (the ground or the platform) in the first frame, and discard clips whose object trajectory then falls below the support surface. All clips are further checked for hand–object contact, foot sliding, interpenetration, and temporal continuity.

B.2 Data Augmentation

Generative augmentation. For Ground Pickup, we augment the data at the generation stage. We edit the object trajectory of each source motion, e.g., by raising or lowering the carrying height, shifting the start or goal position, or adding lateral detours and turns, and synthesize new human motions conditioned on the edited trajectory with the OMOMO generative model under several random seeds. The generated motions are retargeted in the same way as the source motions, yielding 11 clips per source motion on average, including the source motion itself.

Rigid-body augmentation. For Platform Pickup and Table Carry, we augment the retargeted clips directly. A time-varying planar rigid transform is applied jointly to the human root, the robot root, and the object, while the local joint angles of the human and the robot remain unchanged. The transform combines a global translation of up to ± 0.25 m along each horizontal axis and a global yaw rotation of up to $\pm 25^\circ$ with small local variations, namely offsets of a few centimeters at the start, goal, and midway positions and yaw rotations of 5–12° at the endpoints, which are blended

smoothly along the trajectory. Because the robot and the object share the same transform at every frame, the hand–object relationship is preserved, and the platform of Platform Pickup is moved together with the start of the object trajectory. The time variation of the transform may still introduce foot sliding, so augmented clips that fail the checks above are resampled with reduced local variation. We generate 50 augmented clips per source motion. The source motions themselves are not included in these datasets.

B.3 Task Objects

Both box tasks use the scanned large box of OMOMO, with a bounding box of $34 \times 34 \times 36$ cm. In Platform Pickup, the box starts on a 34×34 cm platform with its top surface between 0.48 and 0.63 m high, placed under the start pose of the box in each clip. Table Carry uses the scanned large table of OMOMO, which is lifted by gripping the edges of its tabletop. Since the captured table size differs across subjects, we scale the mesh per subject, resulting in five sizes with tabletop widths from 38 to 44 cm. We exclude three table motions in which the table is turned on its side rather than carried level. All objects are simulated as rigid bodies of 0.5 kg, with the center of mass and inertia of a uniform thin shell over the object surface.

B.4 Train/Test Splits

For each task, we split the data by subject: all clips derived from the motions of a held-out subject form the test set, and no clip of that subject is used for training. Table 6 summarizes the resulting splits. Ground Pickup trains on three subjects and tests on a fourth; Platform Pickup and Table Carry each hold out one subject whose two source motions yield 100 test clips.

Table 6: Dataset statistics. Each entry lists the training / test counts.

Task	Subjects	Source motions	Clips
Platform Pickup	1 / 1	2 / 2	100 / 100
Ground Pickup	3 / 1	21 / 6	216 / 80
Table Carry	4 / 1	6 / 2	300 / 100

C Baseline and Ablation Implementation

Common setup. All methods are evaluated in the same Isaac Gym environment as our policy, at a control frequency of 50 Hz, on the same test clips and with the same object model (Appendix B.3). Each episode starts from the first frame of the reference: the robot is initialized to the reference pose and velocity with its feet on the ground, and the object rests on its support surface. Domain randomization and observation noise are disabled, and each clip is evaluated once in a separate simulator instance, so that the results do not depend on the evaluation order. The baselines are used as released, without fine-tuning on our data, and none of them observes the object. All metrics are computed with the same criteria as for ARIA (Appendix D).

SONIC. We use the released SONIC model (Luo et al., 2025), which also serves as the frozen base policy π_{base} in our framework. It receives the human motion of each clip as the operator command g_t^h, as in ARIA, and its output is executed without correction, i.e., $\Delta a_t^{\text{res}} = 0$. The comparison with SONIC therefore shows the effect of adding the assistive layer to the same base policy.

HoloMotion. HoloMotion (Chen et al., 2026) is a general motion tracking controller, implemented as a mixture-of-experts Transformer conditioned on the current and the next 10 frames of a reference robot motion; we use the released v1.4.1 model. Since it tracks robot rather than human motion, we provide the retargeted G1 motion of each clip, resampled to 50 Hz, i.e., the same trajectory that serves as the tracking reference of our residual policy. This reference is obtained by interaction-aware retargeting and already preserves the hand–object relationship of the demonstration (Appendix B.1), so HoloMotion and TeleopIt receive a more object-consistent command than the human motion given to SONIC.

TeleopIt. TeleopIt (Wu et al., 2026a) is a whole-body teleoperation system whose tracking policy is a temporal convolutional network over a 10-frame observation history, conditioned on the current frame of a retargeted robot motion; we use its released general tracking policy (v0.5.0). It receives the same retargeted G1 motion as HoloMotion, interpolated at each control step, without the reference smoothing used for live teleoperation. For both HoloMotion and TeleopIt, we use the PD gains, action scaling, and default joint pose of the released models, and reimplement their observation construction in our environment, which we verify numerically against the official implementations.

Ablations. All ablations are evaluated with the same protocol as ARIA. Fine-tuned SONIC updates the action decoder of the released SONIC model together with its action noise, while its encoders stay frozen. Starting from the released weights, it is trained with PPO on the same training clips and reward terms as ARIA, using the optimization settings of SONIC (adaptive learning rate within $[10^{-5}, 2 \times 10^{-4}]$, maximum gradient norm 0.1, and entropy coefficient 0.01); a new critic with the architecture of Appendix A.2 is trained alone for the first 100 iterations. The w/o geometry variant is trained with the same recipe as ARIA, with s_t^o set to zero during training and evaluation; the perception gate still uses the hand-to-object distances. The w/o gate variant evaluates the trained ARIA policy with $\eta_t \equiv 1$.

D Success Criteria and Breakdown

D.1 Success Criteria

An episode is evaluated over the full length of its reference clip and counts as a success only if all three conditions below hold. Object heights are measured at the lowest point of the object's surface point cloud rather than at its origin, so that tipping or flipping the object over is not mistaken for lifting it.

No Fall. Throughout the episode, the pelvis height stays above $\max(0.25\,\mathrm{m},\, 0.5\,h_0)$, where h_0 is the initial pelvis height, and the absolute roll and pitch of the pelvis stay below 1.2 rad.

Lift. The lowest point of the object rises at least δ above its initial height at some time during the episode, with $\delta = 0.05\,\mathrm{m}$ for Platform Pickup and $\delta = 0.20\,\mathrm{m}$ for Ground Pickup and Table Carry. The threshold is smaller for Platform Pickup because the box starts on an elevated platform rather than on the ground.

Stable Hold. The hold duration is measured from the first frame at which the lift threshold is reached until the object returns to the ground (its lowest point falls below 0.15 m) or the episode ends. The same quantity is computed on the reference motion, and the condition holds if the simulated hold duration is at least half of the reference hold duration.

D.2 Success Breakdown

Table 7 reports the fraction of test episodes that satisfy each condition. All methods keep the robot balanced in almost every episode, so the failures of the baselines stem from object interaction rather than from falling. The baselines rarely lift the object at all; the few lifts they achieve (3.0% for SONIC on Platform Pickup and 15.0% for HoloMotion on Ground Pickup) are brief and never satisfy the hold condition. ARIA lifts the object in 100.0%, 53.8%, and 100.0% of the episodes on the three tasks, and almost all of these lifts are also held, so that SR is close to the lift rate. The remaining failures of ARIA, on Ground Pickup, are therefore mainly failures to lift the object rather than drops after lifting.

E Real-Robot System

E.1 Hardware and Compute

We deploy ARIA on a Unitree G1 humanoid with 29 actuated joints; as in simulation, the correction acts only on the 14 arm joints. Object perception uses an Intel RealSense D435i mounted on the head of the robot, of which we use the left infrared stream at 848×480 pixels and 30 Hz. Computation is split between the robot and an external workstation connected by wired Ethernet. The

Table 7: Breakdown of the success rate into its three conditions (%). NF: no fall; Lift: object lifted; Hold: lift maintained (which implies Lift); SR: all three conditions hold.

	Platform Pickup				Ground Pickup				Table Carry			
Method	NF	Lift	Hold	SR	NF	Lift	Hold	SR	NF	Lift	Hold	SR
SONIC	100.0	3.0	0.0	0.0	100.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0
HoloMotion	100.0	0.0	0.0	0.0	98.8	15.0	0.0	0.0	100.0	0.0	0.0	0.0
TeleopIt	100.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0
ARIA	100.0	**100.0**	**100.0**	**100.0**	100.0	**53.8**	**52.5**	**52.5**	100.0	**100.0**	**100.0**	**100.0**

onboard NVIDIA Jetson Orin NX only captures and streams the camera images, while object pose estimation, the processing of the operator's motion, and both policies run on a workstation with an Intel Core i7-12700KF CPU and an NVIDIA RTX 3090 GPU; π_{base} is executed with TensorRT, and π_{res} is exported to ONNX. The operator wears a PICO 4 Ultra headset and body trackers, whose whole-body tracking, accessed through XRoboToolkit (Zhao et al., 2025b), streams the poses of 24 body joints at about 90 Hz. These poses are converted into the SMPL motion command g_t^h of π_{base} (Eq. 2) at 50 Hz. Alternatively, the operator wears an inertial motion-capture suit (Zuo et al., 2025), calibrated in a T-pose before each session, from whose ten body-worn IMUs the workstation estimates a 24-joint SMPL pose at 30 Hz, resampled to g_t^h at 50 Hz. Only one of the two sources is used per session, and all other components are shared.

E.2 Marker Layout and Calibration

The box has nominal dimensions of $30 \times 30 \times 40$ cm, and its six faces carry 66 ArUco markers (Garrido-Jurado et al., 2014) with a side length of 80 mm: 12 on each side face and 9 on the top and bottom faces. The corner positions of all markers are known in the object frame, whose origin is at the center of the box. In each image, we detect the markers and estimate the pose $T_t^{o|C}$ of the box in the camera frame C by PnP over the detected corners of each face. A face yields a pose candidate when at least four of its markers are detected, and the candidate is accepted only if it lies in front of the camera, has a small reprojection error, and agrees with the markers left out of the fit; a single visible face thus suffices. The camera intrinsics are the factory calibration of the infrared stream. The pose of the box in the pelvis frame P is obtained through the kinematic chain

$$T_t^{o|P} = T^{H|P}(q_t)\, T^{C|H}\, T_t^{o|C}, \tag{9}$$

where H is the head link, $T^{H|P}(q_t)$ is given by forward kinematics with the joint positions at the image acquisition time, and $T^{C|H}$ is the fixed mounting transform of the camera. We take $T^{C|H}$ from the manufacturer's nominal robot model, in which the camera is pitched down by $47.6°$, and check its orientation against the measured gravity direction; the full transform is not calibrated further. From $T_t^{o|P}$ and the hand poses given by forward kinematics, the object geometry s_t^o (Eq. 6) is computed as in simulation. Since the policy is trained with the scanned box of Appendix B.3, the surface distances d_t^h are computed with the surface points of its mesh, aligned with the center and axes of the estimated box pose.

E.3 Runtime Pipeline and Latency

The runtime consists of a teleoperation branch and a perception branch. In the teleoperation branch, the operator's body pose is converted into g_t^h and fed to π_{base}, which produces a_t^{base} at 50 Hz. In the perception branch, marker detection and the kinematic chain of Eq. 9 yield s_t^o and the validity flag v_t. π_{res} runs at the same 50 Hz and combines both branches with the robot state into the gated and bounded correction of Eq. 7; it does not observe the camera images directly. The resulting joint targets a_t (Eq. 1) are sent to the joint PD controllers of the robot, and a single command writer republishes the latest target at 500 Hz between policy steps. Image acquisition, pose estimation, and visualization run asynchronously, and each stage keeps only the latest frame, so that stale images do not accumulate. Every pose estimate carries the acquisition time of its image, and, as in the perception error model (Sec. 3.3.2), estimates older than 100 ms are treated as invalid (v_t=0).

Table 8 reports the latencies recorded over a PICO VR teleoperation session of 2,517 camera frames, during which π_{res} was evaluated online without applying its correction. The median age of the object pose estimates, 70 ms, lies within the 20–80 ms delay range modeled in training, and 99.4% of the estimates are younger than the 100 ms validity threshold. The operator reference and the robot state reach π_{res} with median ages of 34 ms and 3 ms. For the motion-capture suit, a separate session gives a median of 29.3 ms (P99 43.6 ms) from packet reception to the estimated pose.

Table 8: Runtime latencies on hardware with PICO VR (ms). Ages are measured from the acquisition of the data; the rows overlap and do not add up to an end-to-end latency.

Quantity	Median	P90
Image transfer from the robot (round trip)	37.9	46.8
Marker detection and pose estimation	7.8	9.8
Age of the object pose estimate	70.5	89.7
Age of the operator reference at π_{res}	34.0	43.2
Age of the robot state at π_{res}	2.5	2.9

E.4 Safety Mechanisms

Bounded correction. In addition to the joint mask (Eq. 3) and the caps of Eq. 7, the correction is limited on hardware to 0.15 rad per joint, and its rate and acceleration to 0.4 rad/s and 10 rad/s^2; the limiter decelerates the correction before it reaches these bounds. These limits act on the correction only and do not constrain a_t^{base}, and they are fixed within a session.

Activation and fallback. The correction is enabled only on an explicit request by the operator, and only while the robot feedback, the operator reference, and the object pose estimate are valid and the observation history of π_{res} is filled; non-finite, stale, and out-of-order inputs are rejected. When the object pose becomes invalid, the correction fades out under the rate and acceleration limits, and teleoperation continues with π_{base} alone; when perception recovers, assistance is not re-enabled automatically but requires a new request by the operator. When the operator reference is older than a source-specific limit (100 ms for PICO VR and 400 ms for the suit), the correction is withdrawn and π_{base} is switched to its built-in balancing mode; teleoperation resumes only after the operator re-arms the system. An operator console provides mode selection and stop controls; with the suit, which has no handheld controllers, the operator requests above are issued from this console. Stop requests take precedence over all other commands.

Command checks. Before being sent to the robot, every command is checked against the joint position limits, including a look-ahead $q_t + h\,\dot{q}_t$ with $h = 40$ ms, and against limits on joint velocity, estimated torque, motor temperature, body tilt, and the freshness of the robot feedback. A violation triggers a damping stop that removes the position stiffness and damps the joint velocities. Since this stop does not keep the robot upright, we treat it as a last resort rather than a recovery behavior.