# **ResMimic**: From General Motion Tracking to Humanoid Whole-Body Loco-Manipulation via Residual Learning

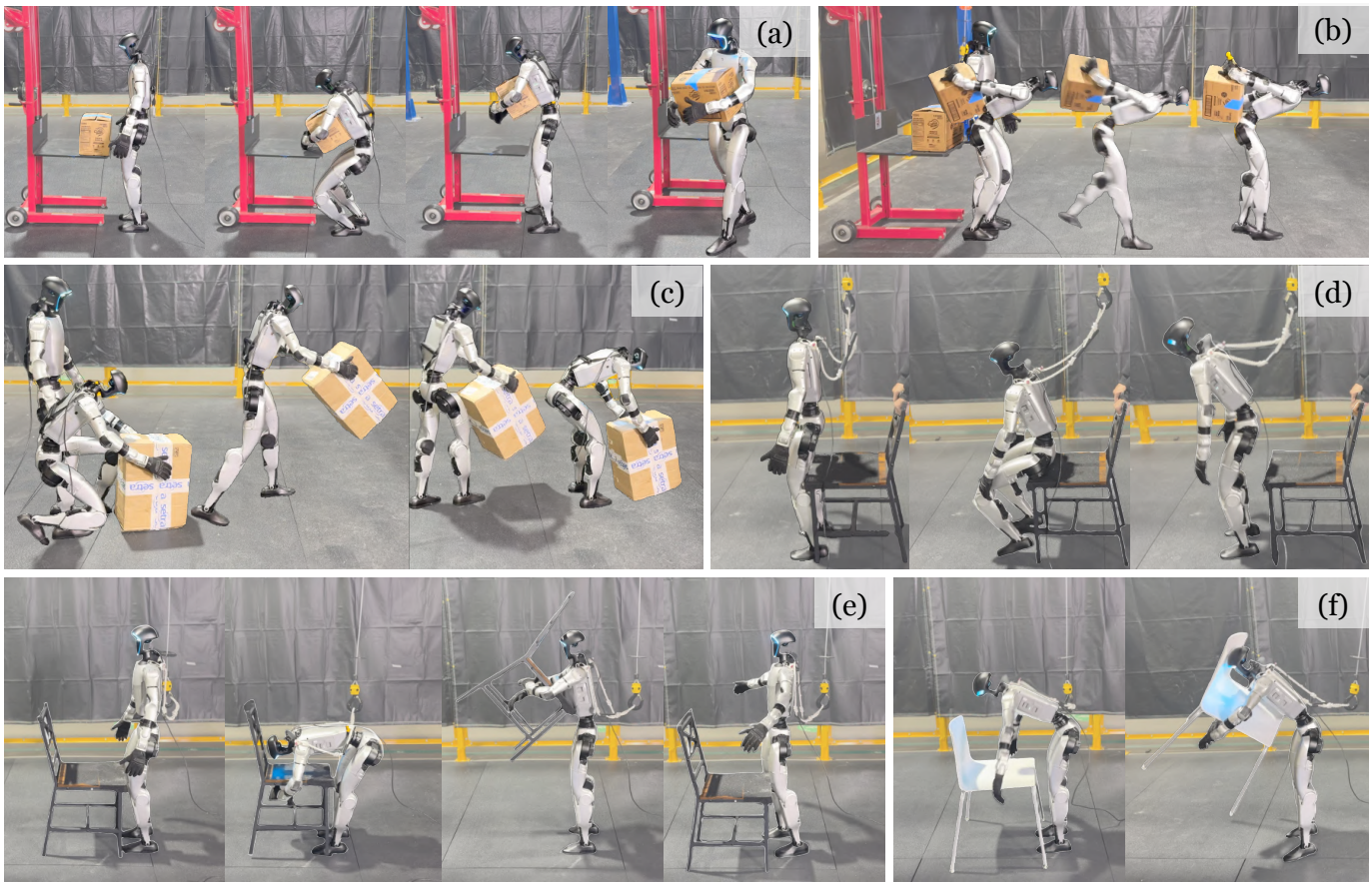


Fig. 1: We deploy `ResMimic` on a Unitree G1 humanoid to demonstrate diverse whole-body loco-manipulation capabilities. (**a**) Carrying 4.5kg heavy payloads via whole-body contact; (**b**, **c**) expressive motion while carrying box; (**d**) general humanoid object interaction beyond manipulation; (**e**, **f**) carrying irregularly shaped heavy objects with instance-level generalization.

***Abstract*—Humanoid whole-body loco-manipulation holds transformative potential in daily service and industrial tasks. While recent advances in general motion tracking (GMT) enable humanoids to reproduce diverse human motions, such policies lack the precision and explicit modeling of object interaction necessary for loco-manipulation. We present `ResMimic`, a two-stage pretrain–post-train residual learning framework for precise humanoid loco-manipulation from human motion data. In the first stage, a GMT policy is pre-trained on large-scale human-only motion data to serve as a task-agnostic whole-body motion prior. In the second stage, a sample-efficient residual policy is post-trained to inject object-conditioned corrections, enabling accurate object interaction without re-learning general motion skills. Our framework employs a unified reward formulation shared across tasks, eliminating per-task reward engineering. To ensure stable and efficient training, we introduce a dense point-cloud object tracking reward, a contact tracking reward, and a virtual force curriculum. We validate `ResMimic` in simulation and on a real Unitree G1 humanoid. Results demonstrate substantial improvements in task success rate, robustness, and data efficiency over strong baselines. Videos and code are available at resmimic-submission.github.io.**

## I. INTRODUCTION

Humanoid robots demonstrate strong potential in real-world applications due to their versatility and human-like embodiment. Unlike quadrupeds or wheeled manipulators, humanoids can perform expressive whole-body loco-manipulation, seamlessly integrating locomotion and manipulation within existing human-centric environments. This capability positions humanoids as a scalable solution for daily service and industrial tasks without modifying infrastructure.

However, achieving precise and scalable humanoid loco-manipulation remains a fundamental challenge. Compared to pure locomotion [28, 26] or tabletop manipulation [17, 34], loco-manipulation requires coordinated whole-body control

under sustained multi-contact interactions, while large-scale interaction data remains difficult to collect. Although direct imitation of human motion [25] is appealing, existing retargeting methods are primarily designed for human-only motion without object interaction. When applied to loco-manipulation, embodiment gaps arise: while object poses remain fixed, relative human–object poses and contact configurations often fail to transfer, leading to failed contact establishment or interpenetration, as shown in Figure 2.

Recent advances in general motion tracking (GMT) [36, 6], trained on large-scale human motion datasets [21, 16], show that humanoids can reproduce diverse motions with high fidelity. However, these policies do not explicitly model object interaction dynamics and typically require teleoperation during manipulation. Conversely, existing loco-manipulation approaches rely heavily on handcrafted pipelines or stage-wise controllers [8, 20], limiting scalability and generality. A unified, data-efficient, and precise framework for humanoid loco-manipulation remains absent.

In parallel, foundation models [24] have demonstrated the effectiveness of large-scale pre-training followed by task-specific post-training. In robotic manipulation, models such as $\pi_0$ [5] and LBM [32] show that this paradigm alleviates data scarcity while improving robustness and generalization. However, such a pretrain–finetune paradigm remains largely unexplored in humanoid whole-body control.

Our key insight is that diverse whole-body motion skills—such as balance recovery, stepping, and reaching—can be learned once as a reusable motion prior through large-scale GMT pre-training, while object-centric loco-manipulation only requires lightweight task-specific corrections. This observation naturally motivates a residual learning paradigm: a stable, general motion prior is post-trained with task-conditioned residual adjustments for object interaction.

To this end, we propose `ResMimic`, a two-stage residual learning framework for humanoid loco-manipulation. In the first stage, a GMT policy is pre-trained on large-scale motion capture data to serve as a robust, task-agnostic prior for human-like whole-body behavior. In the second stage, a sample-efficient residual policy is post-trained to condition on object reference trajectories and output corrective actions that refine the GMT prior for precise manipulation. Importantly, this framework requires only a unified reward formulation shared across tasks, without per-task reward shaping or controller redesign. The decoupling between motion prior and interaction correction improves data efficiency, simplifies system design, and enables scalable adaptation across diverse loco-manipulation tasks. Our contributions are four-fold:

- We introduce a two-stage residual learning paradigm for humanoid loco-manipulation, combining large-scale GMT pre-training with lightweight residual adaptation.
- We design a unified reward formulation—shared across tasks—that eliminates per-task reward engineering, improving scalability and simplicity.
- We propose (i) a point-cloud–based object tracking reward providing dense supervision, (ii) a contact-tracking reward aligned with human demonstrations, and (iii) a curriculum-based virtual object controller to stabilize training and improve sim-to-real transfer.
- We conduct extensive evaluations in simulation and on real hardware, demonstrating significant improvements in motion tracking, object manipulation accuracy, task success rate, training efficiency, robustness, and generalization.

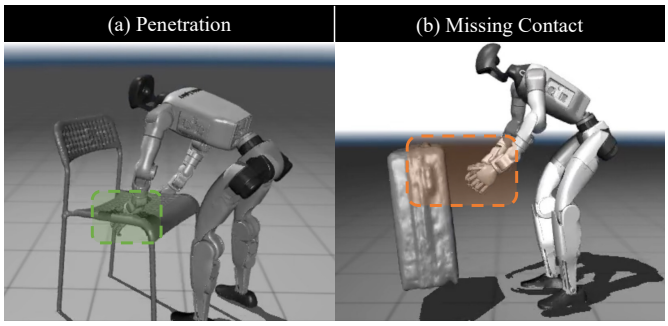


Fig. 2: Visualization of retargeting artifacts caused by embodiment gaps in humanoid–object interaction data: **(a)** hand–chair penetration; **(b)** hand–box missing contact.

## II. RELATED WORK

### A. Learning-Based Humanoid Control

Reinforcement learning (RL) enables real-time whole-body control of humanoids by optimizing policies through environment interaction. However, such approaches often suffer from low data efficiency and require substantial task-specific reward engineering [30]. Consequently, most prior work has focused on locomotion [27, 40] or isolated skills such as recovery [13] and balance stabilization [37], rather than scalable whole-body control across diverse tasks.

To enable more general control, learning from human motions has emerged as a promising direction [25]. These methods typically rely on kinematic retargeting to map human motions to humanoids, bridging the embodiment gap. It enables accurate tracking of individual motions [12, 19, 23] or aims for versatile general motion tracking [7, 11, 36, 6]. While these methods demonstrate whole-body control ability, they remain limited in their ability to interact with physical world. VideoMimic [2] extends motion tracking to contextual scenes by reconstructing static environments, enabling interactions such as sitting. However, it does not address physical manipulation involving dynamic robot–object contact during loco-manipulation. In contrast, our work focuses on contact-rich humanoid loco-manipulation that integrates whole-body motion with physical object interaction.

### B. Humanoid Loco-Manipulation

Humanoid loco-manipulation introduces additional challenges due to the tight coupling of locomotion, manipulation, and sustained multi-contact interaction. Recent work has leveraged teleoperation to collect demonstration data for such tasks [36, 15, 4, 38]. While effective for data collection, these approaches require human operators and do not directly yield scalable, autonomous policies conditioned on object dynamics.

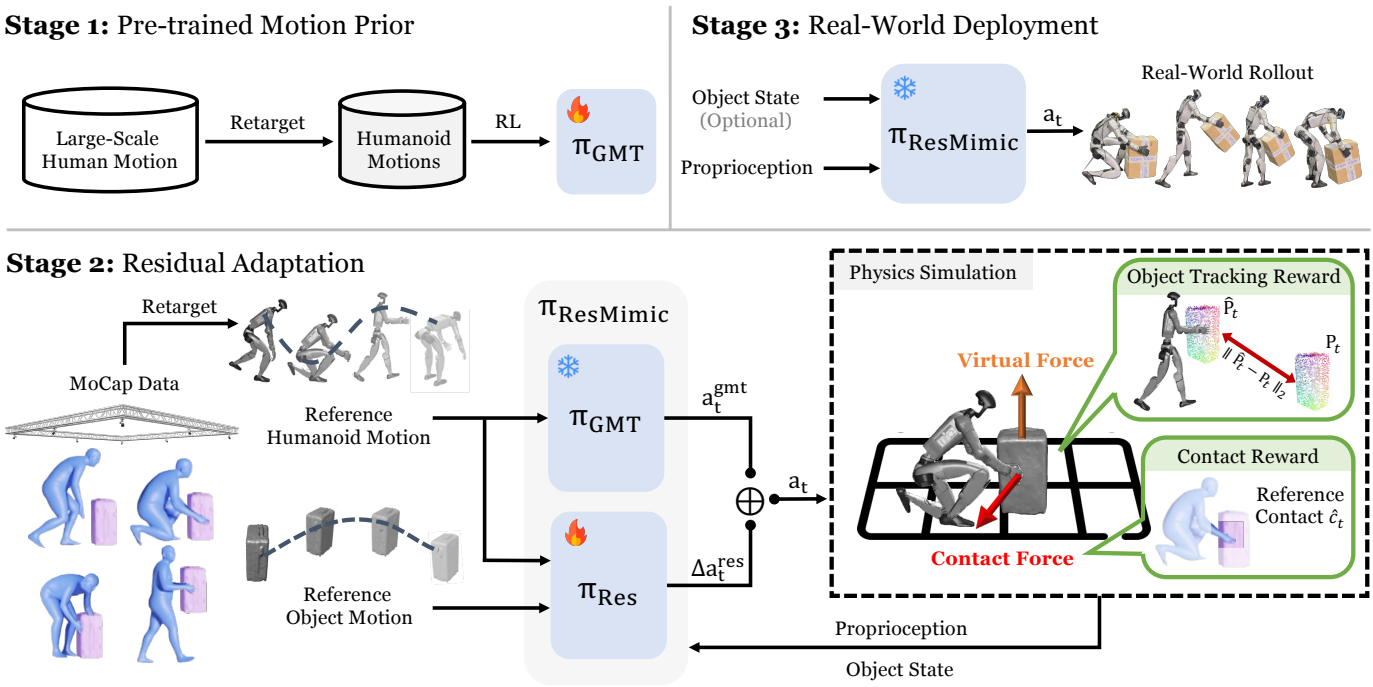


Fig. 3: Overview of `ResMimic`: (1) A general motion tracking policy is trained on large-scale human motion to serve as motion prior. (2) A task-specific residual policy is efficiently trained with virtual force, object and contact reward, to refine the base policy outputs. (3) During real-world deployment, the combined policy is employed for loco-manipulation control.

Building on teleoperated data, imitation learning methods have been developed for autonomous execution [9, 34]. However, these works primarily address tabletop hand-centric manipulation with limited whole-body contact under locomotion. The most closely related works are [8, 20]. Dao *et al.* [8] propose a modular sim-to-real RL framework that decomposes box loco-manipulation into distinct phases with dedicated policies. Liu *et al.* [20] introduce an end-to-end pipeline guided by task-specific trajectory optimization. Despite promising results, these approaches rely on task-specific controller decomposition or reward engineering, limiting scalability across diverse loco-manipulation tasks and demonstrate limited whole-body contact (e.g., using only hands). In contrast, our method maintains a unified architecture and shared reward across tasks by leveraging a pre-trained motion prior with residual task adaptation.

### C. Residual Learning for Robotics

Residual learning has been widely adopted in robot manipulation to refine predefined or learned base policies. Early works introduced residual policies to augment hand-designed controllers or model predictive control, improving precision and contact handling [14, 31]. Subsequent approaches extended residual learning to policies initialized from demonstrations [1, 10]. In dexterous manipulation, residual policies have been used to adapt human motion priors for task-oriented control [18, 39]. In humanoids, ASAP [12] employs residual learning to compensate for sim-to-real dynamics mismatch for agile skill execution. Distinct from them, we leverage a pre-trained general motion tracking policy as a reusable whole-body motion prior and learn residual policies to inject object-conditioned corrections, enabling contact-rich humanoid whole-body loco-manipulation under unified reward formulation and learning framework.

## III. METHOD

We formulate humanoid whole-body loco-manipulation as a goal-conditioned RL problem within a Markov Decision Process (MDP) $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma \rangle$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ the action space, $\mathcal{T}$ the transition dynamics, $\mathcal{R}$ the reward function, and $\gamma$ the discount factor. At time $t$, the state $s_t \in \mathcal{S}$ includes: (i) robot proprioception $s_t^r$, (ii) object state $s_t^o$, (iii) motion goal state $\hat{s}_t^r$, and (iv) object goal state $\hat{s}_t^o$. The action $a_t$ specifies target joint angles, which are executed on the robot through a PD controller. The reward is defined as $r_t = \mathcal{R}(s_t, a_t)$, while the training objective is to maximize the expected cumulative discounted reward: $\mathbb{E}[\sum_{t=1}^{T} \gamma^{t-1} r_t]$.

### A. Two-Stage Residual Learning

To learn humanoid whole-body loco-manipulation from human–object interaction data in a scalable manner, we adopt a two-stage pretrain–post-train residual learning framework, as shown in Figure 3. The key idea is to decouple reusable whole-body motion skills from object-conditioned task adaptation. **Stage I: Pre-trained Motion Prior.** We first train a general motion tracking policy $\pi_{\text{GMT}}$ on large-scale human motion data as motion prior. Given robot proprioception $s_t^r$ and reference motion $\hat{s}_t^r$, the policy outputs a prior action $a_t^{\text{gmt}} = \pi_{\text{GMT}}(s_t^r, \hat{s}_t^r)$, which is optimized to maximize the motion tracking reward $\mathbb{E}\big[\sum_{t=1}^{T} \gamma^{t-1} r_t^m\big]$. **Stage II: Residual Adaptation.** Building on the frozen motion prior, we

train an object-conditioned residual policy $\pi_{\text{Res}}$ under shared architecture and unified reward. The residual policy refines the prior action using both robot and object information: $\pi_{\text{Res}}(s_t^r, s_t^o, \hat{s}_t^r, \hat{s}_t^o) = \Delta a_t^{\text{res}}$. The final action is computed as $a_t = a_t^{\text{gmt}} + \Delta a_t^{\text{res}}$. The residual policy is optimized to maximize the combined rewards $\mathbb{E}\big[\sum_{t=1}^{T} \gamma^{t-1}(r_t^m + r_t^o)\big]$. Both stages are trained using PPO [29].

### B. Pre-trained Motion Prior

In this stage, we train a general motion tracking policy $\pi_{\text{GMT}}$ that takes humanoid proprioception $s_t^r$ and human reference motion $\hat{s}_t^r$ as input, and outputs actions for the humanoid to track the reference motion. The resulting policy serves as a reusable whole-body motion prior for downstream loco-manipulation tasks. While `ResMimic` is a general framework that can incorporate different GMT implementations, we describe one representative instantiation below.

*1) Dataset:* An important motivation of our two-stage training pipeline is to decouple human motion tracking from object interaction. The general motion tracking policy relies solely on human motion capture data, avoiding the need for costly and hard-to-obtain manipulation data.

We leverage several publicly available MoCap datasets for GMT pre-training, including AMASS [21] and OMOMO [16], which together contain over 15,000 clips (approximately 42 hours). After curating the human motion dataset, we apply kinematics-based motion retargeting (e.g. GMR [35]) to transfer human motions into a humanoid reference motion dataset $\{\hat{S}_i^r = \{\hat{s}_t^r\}_{t=1}^T\}_{i=1}^D$.

*2) Training Strategy:* To train a real-world deployable general motion tracking policy $\pi_{\text{GMT}}$, we adopt a single-stage RL framework in simulation without access to privileged information. The proprioceptive observation $s_t^r$ is defined as $[\theta_t, \omega_t, q_t, \dot{q}_t, a_t^{\text{hist}}]_{t-10:t}$, where $\theta_t$ is the root orientation, $\omega_t$ is the root angular velocity, $q_t \in \mathbb{R}^{29}$ is the joint position, $\dot{q}_t$ is the joint velocity, and $a_t^{\text{hist}}$ is the recent action history. The reference motion input $\hat{s}_t^r$ is defined as $[\hat{p}_t, \hat{\theta}_t, \hat{q}_t]_{t-10:t+10}$, where $\hat{p}_t$ is the reference root translation, $\hat{\theta}_t$ is the reference root orientation, and $\hat{q}_t$ is the reference joint position. To improve tracking quality, we also incorporate future reference motion into the input, enabling the policy to anticipate and plan for upcoming targets, which yields smoother tracking.

*3) Reward and Domain Randomization:* Following TWIST [36], the motion tracking reward $r_t^m$ is formulated as the sum of three components: (i) task rewards, (ii) penalty terms, and (iii) regularization terms. To promote robust and generalizable sim-to-real transfer, we further apply domain randomization during training. See [36] for more details.

### C. Residual Adaptation Policy

Building on the pre-trained motion prior $\pi_{\text{GMT}}$, we introduce an object-conditioned residual policy $\pi_{\text{Res}}$ to adapt the prior actions for task-specific object interaction.

*1) Reference Motions:* We obtain reference trajectories using a MoCap system that simultaneously records human motion $\{\hat{h}_t\}_{t=1}^T$ and object motion $\{\hat{o}_t\}_{t=1}^T$. The human motion is retargeted to the humanoid robot using GMR [35], yielding humanoid reference trajectories $\{\hat{s}_t^r = \text{GMR}(\hat{h}_t)\}_{t=1}^T$. The object motion is directly used as the reference $\hat{s}_t^o$. Together, these form the complete reference trajectory $\{(\hat{s}_t^r, \hat{s}_t^o)\}_{t=1}^T$ for training the residual policy.

*2) Training Strategy:* We adopt single-stage RL with PPO for residual learning. The residual policy $\pi_{\text{Res}}$ takes $\langle s_t^r, s_t^o, \hat{s}_t^r, \hat{s}_t^o \rangle$ as input and outputs a residual action $\Delta a_t^{\text{res}} \in \mathbb{R}^{29}$. The object state is represented as $s_t^o = [p_t^o, \theta_t^o, v_t^o, \omega_t^o]$, and the reference object trajectory as $\hat{s}_t^o = [\hat{p}_t^o, \hat{\theta}_t^o, \hat{v}_t^o, \hat{\omega}_t^o]_{t-10:t+10}$, where $p_t^o$ denotes the object root translation, $\theta_t^o$ the root orientation, $v_t^o$ the root velocity, and $\omega_t^o$ the root angular velocity.

**Network Initialization.** At the start of training, the humanoid already closely mimics the reference human motion, so the residual policy should ideally output values near zero. To enforce this, we initialize the final layer of the PPO actor using Xavier uniform initialization with a small gain factor (i.e., a scalar that scales the variance of the initialized weights) so that the initial outputs are close to zero [3].

**Virtual Force Curriculum.** While the two-stage residual framework performs well on simple tasks with reliable reference trajectories and light objects, it often fails early when reference motions are noisy or objects are heavy. Failures arise mainly from (i) penetration introduced by kinematic retargeting, which causes the humanoid to push or knock over the object when naively imitating the reference–leading the initial policy to retreat rather than engage–and (ii) instability when handling large object masses.

To address this, inspired by [22], we introduce a virtual force curriculum that stabilizes training by driving the object toward its reference trajectory. At each timestep, PD controllers apply virtual forces and torques:

$$\mathcal{F}_t = k_p(\hat{p}_t^o - p_t^o) - k_d v_t^o, \quad \mathcal{T}_t = k_p(\hat{\theta}_t^o \ominus \theta_t^o) - k_d \omega_t^o,$$

where $\mathcal{F}_t$ and $\mathcal{T}_t$ are the control force and torque, and $\ominus$ denotes the rotation difference. The controller gains $(k_p, k_d)$ are gradually decayed, so that early training is stabilized by strong virtual assistance, while later training forces the policy to take over and complete the task autonomously.

*3) Reward and Early Termination:* Decoupling motion tracking from object interaction brings an additional advantage: we avoid carefully tuning the relative weights between motion and object rewards. Instead, we directly reuse the motion reward $r_t^m$ and domain randomization from GMT training, and introduce two additional terms: the object tracking reward $r_t^o$, which encourages task completion, and the contact tracking reward $r_t^c$, which provides explicit guidance on body–object contact, improving real-world deployability.

**Object Tracking Reward.** Prior work [20, 33] typically measures object tracking using pose differences between the simulated and reference objects, e.g., $r_t^o = \exp(-\lambda_p \|p_t^o - \hat{p}_t^o\|_2) + w \cdot \exp(-\lambda_\theta \|\theta_t^o \ominus \hat{\theta}_t^o\|_2)$. Instead, we propose an alternative with smoother reward landscape: sample $N$ points from the object mesh surface and compute the point-cloud

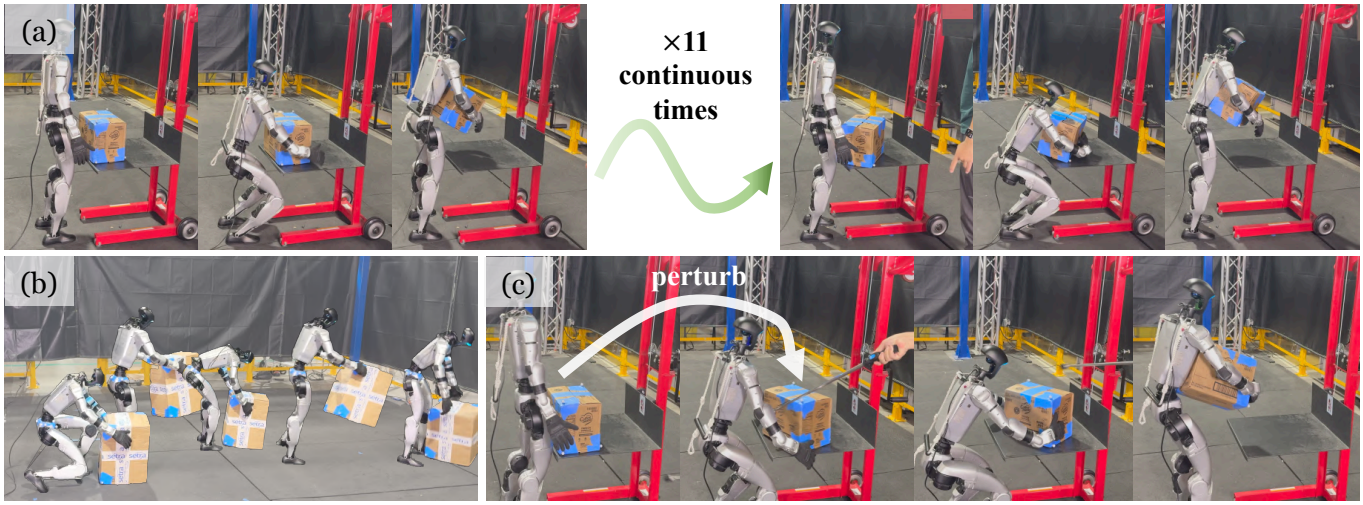


Fig. 4: We deploy `ResMimic` on Unitree G1 with MoCap-based object states. (**a**) Lifting a box from random object initial poses across 11 trials; (**b**) Autonomous consecutive kneeling and box lifting; (**c**) Reactive behavior to external perturbations.

difference between the current and reference states,

$$r_t^o = \exp(-\lambda_o \sum_{i=1}^{N} \|\mathbf{P}[i]_t - \hat{\mathbf{P}}[i]_t\|_2),$$

where $\mathbf{P}_t \in \mathbb{R}^{N\times 3}$ denotes the sampled 3D points. This approach naturally accounts for both translation and rotation, eliminating the need for task-specific weight tuning.

**Contact Tracking Reward.** To encourage correct physical interactions during whole-body manipulation while remaining efficient, we discretize contact locations into meaningful links, e.g., torso, hip, and arms, excluding feet since they mainly contact the ground. Oracle contact information is obtained from the reference human–object interaction data:

$$\hat{c}_t[i] = \mathbf{1}(\|\hat{d}_t[i]\| < \sigma_c),$$

where $i$ indexes the links, $\mathbf{1}(\cdot)$ is the indicator function, and $\|\hat{d}_t[i]\|$ is the distance between link $i$ and the object surface. The contact tracking reward is then defined as

$$r_t^c = \sum_i \hat{c}_t[i] \cdot \exp\Big(-\frac{\lambda}{f_t[i]}\Big),$$

where $f_t[i]$ is the contact force at link $i$.

**Early Termination.** Commonly used in motion tracking, early termination [25] ends an episode if a body part makes unintended ground contact or deviates substantially from the reference, preventing the policy from overvaluing invalid states. For humanoid whole-body loco-manipulation, we introduce additional conditions: (i) the object mesh deviates from its reference beyond a threshold, $\|\mathbf{P}_t - \hat{\mathbf{P}}_t\|_2 > \sigma_o$, or (ii) any required body–object contact is lost for more than 10 consecutive frames.

## IV. EXPERIMENTS

We evaluate `ResMimic` through a combination of large-scale simulation studies and real-world deployment on a Unitree G1 humanoid robot (29-DoF, 1.3 m tall). The evaluation is designed to investigate both algorithmic efficiency and deployment robustness. In particular, we focus on the following research questions:

**Q1:** Can a general motion tracking policy perform diverse loco-manipulation tasks without task-specific retraining?

**Q2:** Does leveraging a pre-trained GMT policy improve sample efficiency and final task performance compared to training from scratch under identical reward settings?

**Q3:** When adapting GMT policies to loco-manipulation tasks, is residual learning more effective than fine-tuning?

**Q4:** Beyond simulation, can `ResMimic` achieve stable, precise, contact-rich loco-manipulation in the real-world?

TABLE I: Sim-to-Sim evaluation of `ResMimic` against baseline approaches in MuJoCo across all four tasks.

| Method | Task | *SR* ↑ | *Iter.* ↓ | $E_o$ ↓ | $E_m$ ↓ | $E_j$ ↓ |
|---|---|---|---|---|---|---|
| Base Policy | Kneel | 0% | – | 0.76 ± 0.01 | 3.30 ± 0.53 | 0.28 ± 0.01 |
| | Carry | 0% | – | 0.29 ± 0.02 | 2.47 ± 0.26 | 1.19 ± 0.30 |
| | Squat | 40% | – | 0.19 ± 0.01 | 0.93 ± 0.07 | 0.90 ± 0.08 |
| | Chair | 0% | – | 1.19 ± 0.48 | 30.18 ± 33.45 | 1.20 ± 0.23 |
| | Mean | 10% | – | 0.61 | 9.22 | **0.89** |
| Train from Scratch | Kneel | 0% | × | 0.69 ± 0.00 | 5.20 ± 0.62 | 3.41 ± 0.07 |
| | Carry | 0% | 6500 | 0.70 ± 0.03 | 5.39 ± 0.38 | 2.33 ± 0.06 |
| | Squat | 0% | 5000 | 0.68 ± 0.05 | 7.56 ± 2.31 | 4.28 ± 0.70 |
| | Chair | 0% | 2000 | 0.97 ± 0.08 | 10.01 ± 1.28 | 13.36 ± 0.92 |
| | Mean | 0% | 4500 | 0.76 | 7.04 | 5.84 |
| Finetune | Kneel | 0% | × | 0.87 ± 0.01 | 5.92 ± 0.81 | 3.02 ± 0.18 |
| | Carry | 30% | 4500 | 0.33 ± 0.01 | 2.49 ± 0.18 | 2.39 ± 0.06 |
| | Squat | 0% | 2000 | 0.47 ± 0.05 | 5.07 ± 1.06 | 2.53 ± 0.13 |
| | Chair | 0% | 700 | 0.15 ± 0.01 | 0.28 ± 0.05 | 1.26 ± 0.09 |
| | Mean | 7.5% | 2400 | 0.46 | 3.44 | 2.30 |
| ResMimic (Ours) | Kneel | 90% | 2000 | 0.14 ± 0.00 | 0.23 ± 0.06 | 2.17 ± 0.06 |
| | Carry | 100% | 1000 | 0.11 ± 0.00 | 0.08 ± 0.00 | 1.24 ± 0.03 |
| | Squat | 80% | 1500 | 0.07 ± 0.01 | 0.07 ± 0.03 | 1.18 ± 0.03 |
| | Chair | 100% | 700 | 0.16 ± 0.01 | 0.13 ± 0.02 | 0.55 ± 0.01 |
| | Mean | **92.5%** | **1300** | **0.12** | **0.13** | 1.29 |

### A. Experiment Setup

*1) Tasks:* We design four challenging whole-body loco-manipulation tasks that stress different aspects of humanoid control and generalization. Task (i) *Kneel on one knee and*

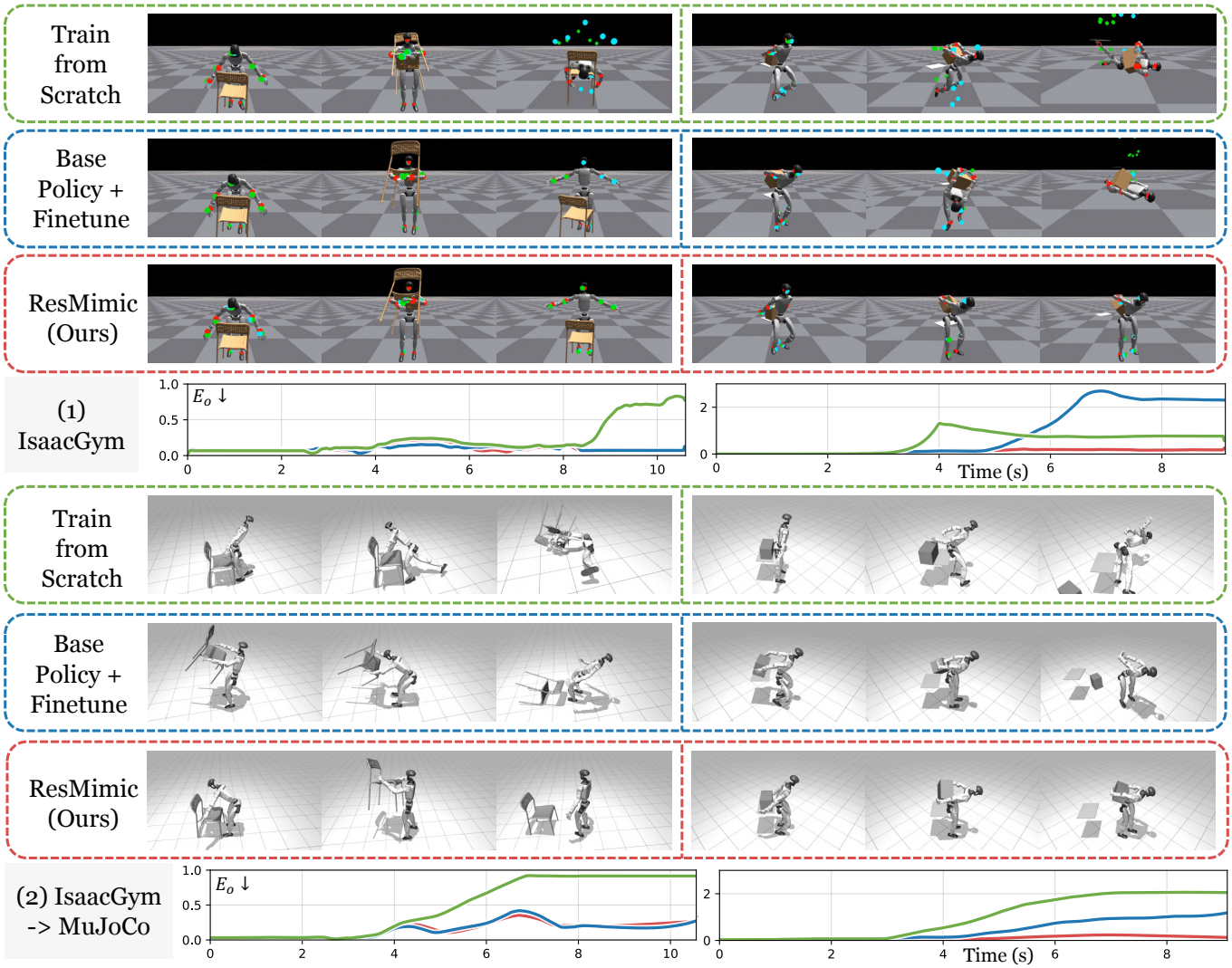


Fig. 5: Comparison between IsaacGym and MuJoCo results for task *Chair* (left) and *Carry* (right). Corresponding curves quantify object tracking error for Train from Scratch, Finetune, and ResMimic.

*lift a box* — requires expressive, large-amplitude motion and precise lower-body coordination. (ii) *Carry a box onto the back* — demands whole-body expressiveness while maintaining balance under shifting load distribution. (iii) *Squat and lift a box with arms and torso* — highlights the challenge of whole-body contact-rich manipulation. (iv) *Lift up a chair* — involves manipulating a heavy, irregularly shaped object. For clarity, we will refer to these tasks as *Kneel*, *Carry*, *Squat*, and *Chair*, respectively. The human–object interaction reference motions for these tasks are collected using OptiTrack motion capture system.

*2) Evaluation Metrics:* We evaluate `ResMimic` in terms of training efficiency, motion fidelity, manipulation accuracy, and overall task completion. The following metrics are used throughout our experiments: (i) **Training Iterations (*Iter.*)**: To eliminate discrepancies due to hardware differences, we report convergence speed in terms of training iterations rather than wall-clock time. Here we determine convergence when reward stops increasing approximately. (ii) **Object Tracking Error**: $E_o = \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{N}||\mathbf{P}[i]_t - \hat{\mathbf{P}}[i]_t||_2$, where $\mathbf{P}_t$ and $\hat{\mathbf{P}}_t$ denote sampled point clouds on the manipulated object mesh and its reference at time $t$. (iii) **Motion Tracking Error**: $E_m = \frac{1}{T}\sum_{t=1}^{T}\sum_{i}||p_t[i] - \hat{p}_t[i]||_2$, where $p_t[i]$ and $\hat{p}_t[i]$ are the global positions of link $i$ at time $t$. (iv) **Joint Tracking Error**: $E_j = \frac{1}{T}\sum_{t=1}^{T}||q_t - \hat{q}_t||_2$, where $q_t, \hat{q}_t \in \mathbb{R}^{29}$ denote the robot's joint angles and their reference values. This evaluates joint-level precision. (v) **Task Success Rate (*SR*)**: A rollout is considered successful if $E_o$ is below a predefined threshold and the robot remains balanced.

*3) Baselines:* To validate the effectiveness and efficiency of our pipeline, we compare `ResMimic` against three representative and strong baselines:

(i) **Base Policy**: The pre-trained GMT policy is directly deployed to track the retargeted human reference motion without conditioning on object states.

(ii) **Train from Scratch**: A single-stage RL policy is trained from scratch to jointly track human motion and object trajectories, without leveraging a pre-trained motion prior.

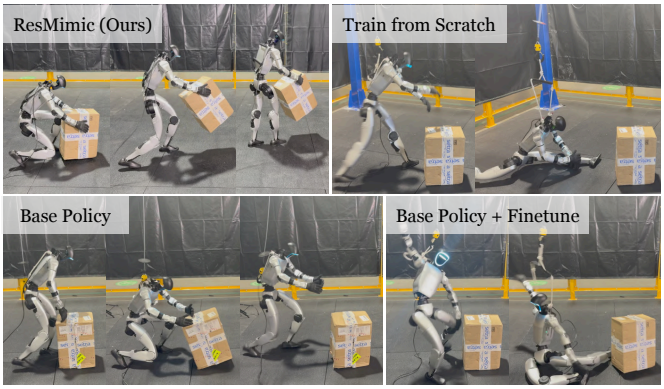


Fig. 6: Real-world qualitative results comparing ResMimic against all other baselines.

The network architecture, observation space, and reward formulation are kept identical to ResMimic to ensure a controlled comparison.

(iii) **Base Policy + Finetune**: The pre-trained GMT policy is fine-tuned under the same reward formulation to track both human motion and object trajectories.

### B. Sim-to-Sim Evaluation

All policies are trained in IsaacGym to leverage its massive parallelism for accelerated data collection. To evaluate generalization, however, we perform sim-to-sim transfer into MuJoCo, which is widely regarded as a better proxy for real-world physics and thus a closer benchmark for sim-to-real performance. We report quantitative results for all baselines across the four loco-manipulation tasks in Table I, and highlight key takeaways below.

**(Q1) GMT alone cannot complete loco-manipulation tasks, but provides a strong initialization.** We first compare ResMimic against directly deploying the pre-trained GMT policy. As shown in Table I, the base GMT policy achieves only a 10% success rate, compared to 92.5% for ResMimic. Although GMT alone yields slightly lower joint tracking error (due to its training objective), it performs poorly on object tracking and overall task completion, as it lacks access to object information. This indicates that while GMT captures joint-level precision, it is insufficient for manipulation without adaptation.

**(Q2) Using GMT as base policy significantly improves training efficiency and effectiveness.** Next, we compare ResMimic with policies trained entirely from scratch under identical settings (reward function, domain randomization, etc.). As shown in Table I, training from scratch fails to solve the tasks in MuJoCo and converges much more slowly. A side-by-side comparison between IsaacGym and MuJoCo as shown in Figure 5 reveals that trained-from-scratch policies sometimes show partial success in IsaacGym but collapse entirely under sim-to-sim transfer. In contrast, ResMimic maintains strong performance with minimal degradation. This demonstrates the necessity of using GMT as a foundation: its large-scale pretraining imbues generalization and robustness against sim-to-sim gaps.

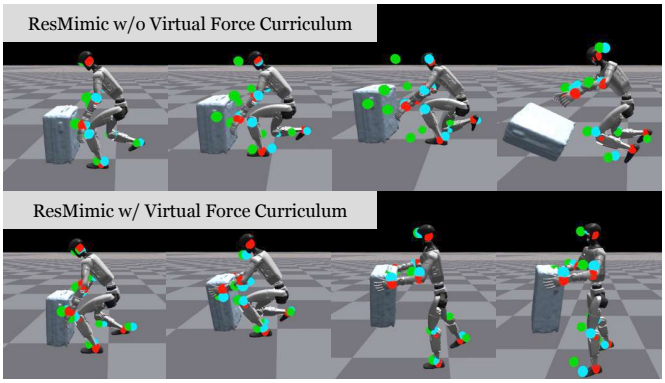


Fig. 7: Ablation on virtual force curriculum.

**(Q3) Residual learning outperforms direct fine-tuning.** Finally, we evaluate residual learning against fine-tuning the GMT policy. While fine-tuning yields slight improvements over training from scratch, it neither outperforms the base GMT policy nor approaches the performance of ResMimic as shown in Table I. A key limitation is that fine-tuning tends to overwrite the generalization capability of the GMT policy, leading to instability across tasks. A comparison between IsaacGym and MuJoCo as shown in Figure 5 reveals that fine-tuned policies succeed on task *Chair-Lift* in IsaacGym but fail to transfer in MuJoCo, underscoring the superiority of residual learning as a more generalizable and extensible adaptation strategy. Moreover, direct fine-tuning operates on the original GMT architecture, which is not explicitly conditioned on object states. As a result, object-tracking rewards alone provide limited supervision for learning robust behaviors, especially under randomized object poses.

### C. Real-world Evaluation

As shown in Figure 1, we deploy ResMimic on a Unitree G1 humanoid and demonstrate precise, expressive, robust whole-body loco-manipulation. ResMimic supports both blind (without object state input) and non-blind deployment (with MoCap-based object state input). For simplicity, all real-world results in Figure 1 are under blind deployment.

- **Expressive carrying motions:** the robot kneels on one knee to pick up a box, or carry the box on its back, highlighting expressive whole-body movement.
- **Humanoid–object interaction beyond manipulation:** the robot sits down on a chair and then stands up while maintaining balance and contact with the environment.
- **Heavy payload carrying with whole-body contact:** the robot successfully carries a 4.5 kg box, while the G1's wrist payload limit is around 2.5 kg, demonstrating the necessity of leveraging whole-body contact.
- **Generalization to irregular heavy objects:** the robot lifts and carries chairs weighing 4.5 kg and 5.5 kg, showing instance-level generalization to novel, non-box geometries.

We also conduct a qualitative comparison of ResMimic against all baselines in the real world, as shown in Figure 6. The results indicate that while the base policy can superficially mimic human motion, it lacks object awareness—an issue that

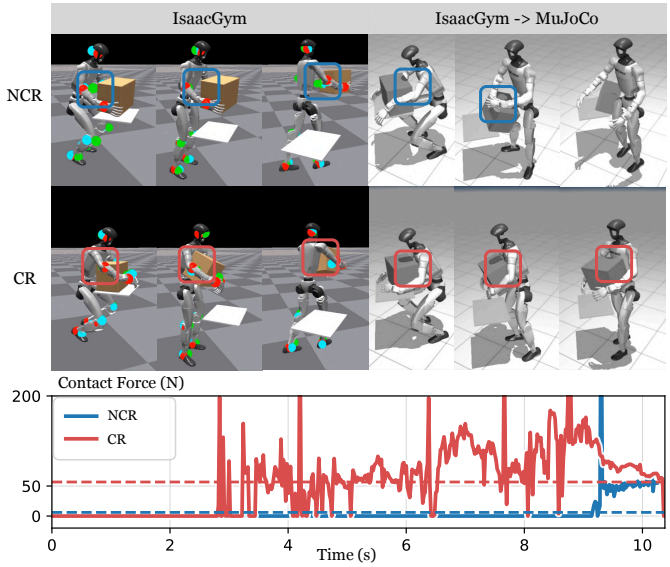


Fig. 8: **Ablation on contact reward.** Here NCR denotes "No Contact Reward", and CR denotes "with Contact Reward". Corresponding curves (bottom) quantify torso contact force.

becomes even more pronounced when the demonstration data is imperfect. Training from scratch and finetuning, on the other hand, fail entirely due to sim-to-real gap.

Finally, we evaluate `ResMimic` under non-blind deployment with MoCap-based object state input as shown in Figure 4. In this setting, the robot demonstrates the ability to (i) manipulate objects from random initial poses, (ii) autonomously perform consecutive loco-manipulation tasks, and (iii) show reactive behavior to external perturbations.

### D. Ablation Studies

*1) Effect of the Virtual Force Curriculum:* The virtual object controller stabilizes early-stage training by applying curriculum-based virtual forces that guide the object toward its reference trajectory. A qualitative example is shown in Figure 7. In this task, the reference motions contain imperfections, including penetrations between the humanoid's hand and the object. The policy initially focuses only on motion tracking to reach the object, causing the object to be knocked over, which yields low object rewards and frequent early terminations. This quickly drives the policy into a local minimum where the robot retreats rather than engaging with the object. In contrast, with the virtual force curriculum, the object remains stabilized during early learning, enabling the policy to overcome motion-data imperfections and converge to precise manipulation strategies.

*2) Effect of the Contact Reward:* The contact reward provides explicit guidance on leveraging whole-body strategies. As illustrated in Figure 8, there are two possible ways to lift the box: (1) relying only on wrists and hands, or (2) engaging both torso and arm contact as demonstrated by humans. Without the contact reward, the policy converges to (1), which may succeed in IsaacGym but fails to transfer to MuJoCo and the real world. With the contact reward, the humanoid instead adopts strategy (2), using coordinated torso and arm contact. This alignment with human demonstrations results in improved sim-to-sim and sim-to-real transfer, validating the importance of the contact reward.

## V. CONCLUSION

We present `ResMimic`, a two-stage pretrain–post-train residual framework for humanoid whole-body loco-manipulation. By learning a reusable whole-body motion prior from large-scale human motion data and refining it with lightweight, object-conditioned residual adaptation, our approach decouples general motion skills from task-specific interaction. The proposed unified reward formulation eliminates per-task reward redesign while maintaining training stability and data efficiency. Extensive experiments in simulation and on a real Unitree G1 humanoid demonstrate consistent improvements in task success, motion fidelity, robustness, and training efficiency. Overall, our results suggest that combining large-scale motion pre-training with structured residual adaptation offers a scalable path toward general-purpose humanoid loco-manipulation.

## REFERENCES

[1] Minttu Alakuijala, Gabriel Dulac-Arnold, Julien Mairal, Jean Ponce, and Cordelia Schmid. Residual reinforcement learning from demonstrations, 2021. URL https://arxiv.org/abs/2106.08050.

[2] 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. *arXiv preprint arXiv:2505.03729*, 2025.

[3] Lars Ankile, Anthony Simeonov, Idan Shenfeld, Marcel Torne, and Pulkit Agrawal. From imitation to refinement – residual rl for precise assembly, 2024. URL https://arxiv.org/abs/2407.16677.

[4] Qingwei Ben, Feiyu Jia, Jia Zeng, Junting Dong, Dahua Lin, and Jiangmiao Pang. Homie: Humanoid loco-manipulation with isomorphic exoskeleton cockpit. *arXiv preprint arXiv:2502.13013*, 2025.

[5] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. $\pi_0$: A vision-language-action flow model for general robot control, 2024. URL https://arxiv.org/abs/2410.24164.

[6] Zixuan Chen, Mazeyu Ji, Xuxin Cheng, Xuanbin Peng, Xue Bin Peng, and Xiaolong Wang. Gmt: General motion tracking for humanoid whole-body control. *arXiv:2506.14770*, 2025.

[7] Xuxin Cheng, Yandong Ji, Junming Chen, Ruihan Yang, Ge Yang, and Xiaolong Wang. Expressive whole-

body control for humanoid robots. *arXiv preprint arXiv:2402.16796*, 2024.

[8] Jeremy Dao, Helei Duan, and Alan Fern. Sim-to-real learning for humanoid box loco-manipulation, 2023. URL https://arxiv.org/abs/2310.03191.

[9] Zipeng Fu, Qingqing Zhao, Qi Wu, Gordon Wetzstein, and Chelsea Finn. Humanplus: Humanoid shadowing and imitation from humans, 2024. URL https://arxiv.org/abs/2406.10454.

[10] Siddhant Haldar, Jyothish Pari, Anant Rai, and Lerrel Pinto. Teach a robot to fish: Versatile imitation from one minute of demonstrations, 2023. URL https://arxiv.org/abs/2303.01497.

[11] 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. *arXiv preprint arXiv:2406.08858*, 2024.

[12] 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. *arXiv preprint arXiv:2502.01143*, 2025.

[13] Tao Huang, Junli Ren, Huayi Wang, Zirui Wang, Qingwei Ben, Muning Wen, Xiao Chen, Jianan Li, and Jiangmiao Pang. Learning humanoid standing-up control across diverse postures. *arXiv preprint arXiv:2502.08378*, 2025.

[14] 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, 2018. URL https://arxiv.org/abs/1812.03201.

[15] Jialong Li, Xuxin Cheng, Tianshu Huang, Shiqi Yang, Rizhao Qiu, and Xiaolong Wang. Amo: Adaptive motion optimization for hyper-dexterous humanoid whole-body control. *Robotics: Science and Systems 2025*, 2025.

[16] Jiaman Li, Jiajun Wu, and C. Karen Liu. Object motion guided human motion synthesis, 2023. URL https://arxiv.org/abs/2309.16237.

[17] Jinhan Li, Yifeng Zhu, Yuqi Xie, Zhenyu Jiang, Mingyo Seo, Georgios Pavlakos, and Yuke Zhu. Okami: Teaching humanoid robots manipulation skills through single video imitation. In *8th Annual Conference on Robot Learning*, 2024.

[18] Kailin Li, Puhao Li, Tengyu Liu, Yuyang Li, and Siyuan Huang. Maniptrans: Efficient dexterous bimanual manipulation transfer via residual learning, 2025. URL https://arxiv.org/abs/2503.21860.

[19] Qiayuan Liao, Takara E. Truong, Xiaoyu Huang, Guy Tevet, Koushil Sreenath, and C. Karen Liu. Beyondmimic: From motion tracking to versatile humanoid control via guided diffusion, 2025. URL https://arxiv.org/abs/2508.08241.

[20] Fukang Liu, Zhaoyuan Gu, Yilin Cai, Ziyi Zhou, Shijie Zhao, Hyunyoung Jung, Sehoon Ha, Yue Chen, Danfei Xu, and Ye Zhao. Opt2skill: Imitating dynamically-feasible whole-body trajectories for versatile humanoid loco-manipulation. *arXiv preprint arXiv:2409.20514*, 2024.

[21] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *International Conference on Computer Vision*, October 2019.

[22] Zhao Mandi, Yifan Hou, Dieter Fox, Yashraj Narang, Ajay Mandlekar, and Shuran Song. Dexmachina: Functional retargeting for bimanual dexterous manipulation. *arXiv preprint arXiv:2505.24853*, 2025.

[23] Jiageng Mao, Siheng Zhao, Siqi Song, Tianheng Shi, Junjie Ye, Mingtong Zhang, Haoran Geng, Jitendra Malik, Vitor Guizilini, and Yue Wang. Learning from massive human videos for universal humanoid pose control, 2024. URL https://arxiv.org/abs/2412.14172.

[24] OpenAI. Gpt-4 technical report, 2024. URL https://arxiv.org/abs/2303.08774.

[25] Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel van de Panne. Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. *ACM Trans. Graph.*, 37(4):143:1–143:14. ISSN 0730-0301. doi: 10.1145/3197517.3201311.

[26] Ilija Radosavovic, Tete Xiao, Bike Zhang, Trevor Darrell, Jitendra Malik, and Koushil Sreenath. Learning humanoid locomotion with transformers. *arXiv:2303.03381*, 2023.

[27] Ilija Radosavovic, Tete Xiao, Bike Zhang, Trevor Darrell, Jitendra Malik, and Koushil Sreenath. Real-world humanoid locomotion with reinforcement learning. *Science Robotics*, 9(89), 2024. doi: 10.1126/scirobotics.adi9579.

[28] Ilija Radosavovic, Bike Zhang, Baifeng Shi, Jathushan Rajasegaran, Sarthak Kamat, Trevor Darrell, Koushil Sreenath, and Jitendra Malik. Humanoid locomotion as next token prediction. *arXiv:2402.19469*, 2024.

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

[30] Carmelo Sferrazza, Dun-Ming Huang, Xingyu Lin, Youngwoon Lee, and Pieter Abbeel. Humanoidbench: Simulated humanoid benchmark for whole-body locomotion and manipulation, 2024. URL https://arxiv.org/abs/2403.10506.

[31] Tom Silver, Kelsey Allen, Josh Tenenbaum, and Leslie Kaelbling. Residual policy learning, 2019. URL https://arxiv.org/abs/1812.06298.

[32] TRI LBM Team. A careful examination of large behavior models for multitask dexterous manipulation, 2025. URL https://arxiv.org/abs/2507.05331.

[33] Sirui Xu, Hung Yu Ling, Yu-Xiong Wang, and Liang-Yan Gui. Intermimic: Towards universal whole-body control for physics-based human-object interactions. *arXiv*

*preprint arXiv:2502.20390*, 2025.

[34] Yanjie Ze, Zixuan Chen, Wenhao Wang, Tianyi Chen, Xialin He, Ying Yuan, Xue Bin Peng, and Jiajun Wu. Generalizable humanoid manipulation with 3d diffusion policies. *arXiv preprint arXiv:2410.10803*, 2024.

[35] Yanjie Ze, João Pedro Araújo, Jiajun Wu, and C. Karen Liu. Gmr: General motion retargeting, 2025. GitHub repository.

[36] 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.

[37] Tong Zhang, Boyuan Zheng, Ruiqian Nai, Yingdong Hu, Yen-Jen Wang, Geng Chen, Fanqi Lin, Jiongye Li, Chuye Hong, Koushil Sreenath, and Yang Gao. Hub: Learning extreme humanoid balance, 2025. URL https://arxiv.org/abs/2505.07294.

[38] Yuanhang Zhang, Yifu Yuan, Prajwal Gurunath, Tairan He, Shayegan Omidshafiei, Ali-akbar Agha-mohammadi, Marcell Vazquez-Chanlatte, Liam Pedersen, and Guanya Shi. Falcon: Learning force-adaptive humanoid loco-manipulation. *arXiv preprint arXiv:2505.06776*, 2025.

[39] Shuqi Zhao, Xinghao Zhu, Yuxin Chen, Chenran Li, Xiang Zhang, Mingyu Ding, and Masayoshi Tomizuka. Dexh2r: Task-oriented dexterous manipulation from human to robots, 2024. URL https://arxiv.org/abs/2411.04428.

[40] Ziwen Zhuang, Shenzhe Yao, and Hang Zhao. Humanoid parkour learning. In *8th Annual Conference on Robot Learning*, 2024.