X-RESET: Scaling Object-Centric Reinforcement Learning via Cross-Embodiment Resets

Prithwish Dan[1,2] Chenyang Ma[1,3†] Wei Zhan[1,4]
[1]Applied Intuition [2]Cornell University [3]University of North Carolina at Chapel Hill
[4]University of California, Berkeley

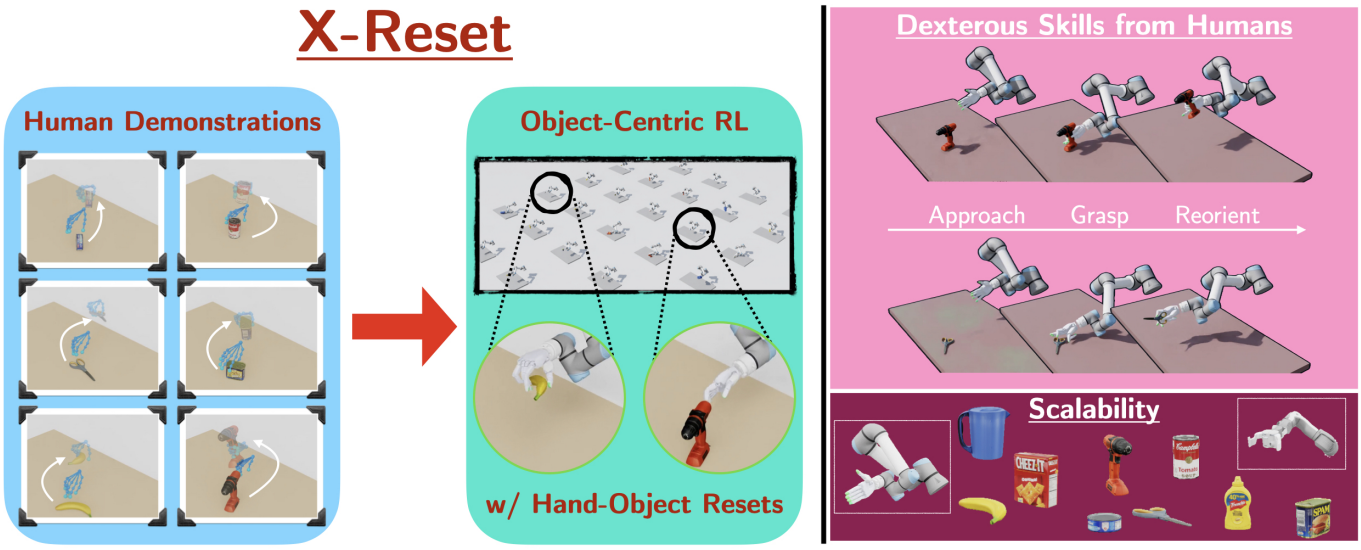

Fig. 1: **Overview of X-RESET:** We introduce X-RESET, a framework to train dexterous manipulation policies by defining reset distributions over human hand-object states during RL training. We define an object-centric reward function to incentivize reorienting objects to goal poses, and incorporate random resets to kinematically retargeted hand-object states to directly expose the policy to success signals. X-RESET can learn the core skills of dexterous manipulation and naturally scales with human demonstrations.

***Abstract*—Human demonstrations are a natural alternative to robot teleoperation for training manipulation policies due to their scale and density of information. Prior works that map human motion to robot actions either make brittle assumptions about equivalent hand kinematics or sidestep the problem by only transferring object motion. We propose X-RESET, a framework to train dexterous manipulation policies by defining reset distributions over human hand-object states during RL training. X-RESET performs kinematic retargeting of hand-object motion to estimate robot trajectories, and randomly samples states as resets during RL training with a generic object-centric reward. In addition, we randomly expose the policy to low-gravity settings where corrective behaviors can more easily be learned. We demonstrate that X-RESET can effectively learn the core manipulation skills—approaching, grasping, and reorienting objects—in an end-to-end fashion, can naturally scale with human demonstrations, and is capable of training various embodiments on diverse objects with a common reward function.**

***Index Terms*—Learning from Human Demonstrations, Dexterous Manipulation, Reinforcement Learning in Simulation**

†Work done during internship at Applied Intuition.

I. INTRODUCTION

The predominant recipe for training robot manipulation policies is imitation learning (IL), in which robot data is used as supervision to learn a mapping from observations to actions. Yet, scaling such approaches to general-purpose models [29, 4] faces the fundamental bottleneck of requiring large-scale datasets [34] of high-quality expert demonstrations. Collecting such data via traditional methods such as teleoperation is time-consuming and expensive, especially when curating for multi-fingered hands with high degrees-of-freedom (DoF). On the other hand, human videos offer a scalable alternative, existing in abundance and encompassing a diverse range of day-to-day hand-object interactions which are critical for manipulation.

Despite recent efforts in building large-scale human manipulation datasets [13, 24, 16], the embodiment gap between humans and robots makes modern IL methods [10, 74] challenging to directly train on this data. Existing works attempt to bridge various segments of the embodiment gap. To obtain robot action labels, one line of work manually defines

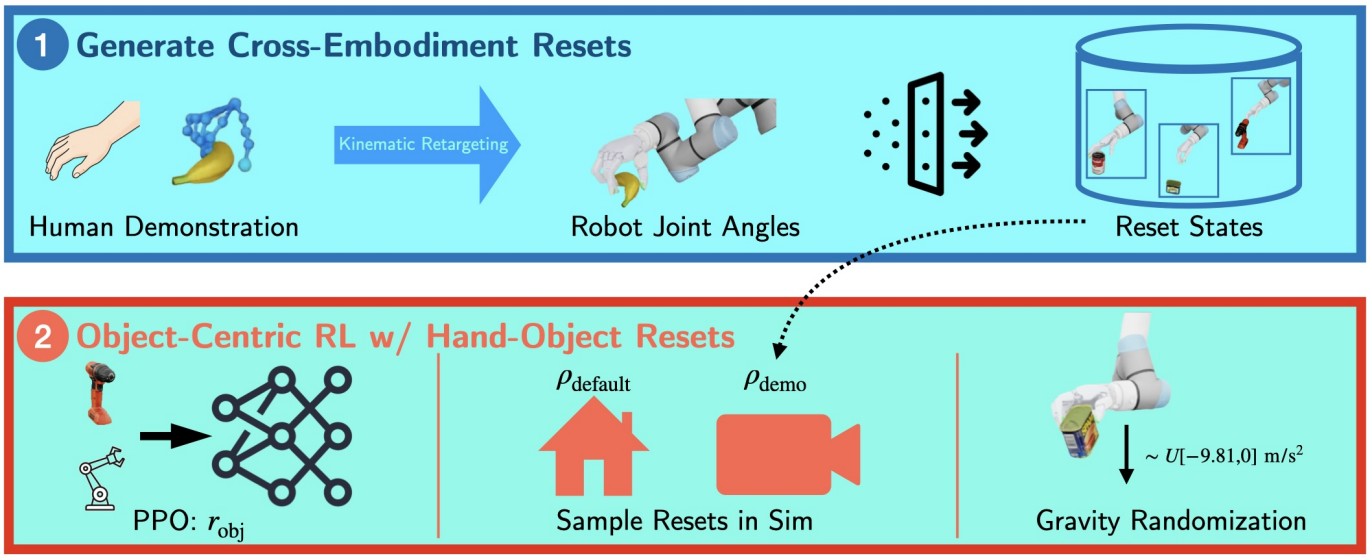

Fig. 2: **Pipeline:** (Top) X-RESET first generates robot hand-object reset states by kinematically retargeting human demonstrations. (Bottom) Then, X-RESET performs object-centric RL training to reorient objects to goal poses, where demonstrated hand-object states are randomly sampled as resets. X-RESET (RANDOM-G) also samples gravity at uniform random to aid in learning from mid-trajectory states

mappings from human hands to robot end-effectors to perform kinematic retargeting [48, 54, 57], while another direction additionally tackles the visual gap [50, 21, 31, 32, 30], for example overlaying robot arms on frames of human videos [31, 32] to use for co-training with real robot data. Such approaches rely on the assumption that human hand actions can directly be imitated on robots, and often fail to obey the true contact dynamics required in robot-object interactions.

A promising approach has been the use of simulation to bridge the embodiment gap [14, 37], leveraging object-centric rewards for reinforcement learning (RL) [9, 14, 37, 20]. Such approaches face a large exploration burden when learning to control many degrees-of-freedom from scratch, requiring additional modules (e.g. motion planning to approach objects [37]) or kinematic motion imitation rewards [73, 66] that must be carefully tuned for different robots and scale poorly as the embodiment gap varies.

We aim to learn the core skills of object manipulation—approaching, grasping, and reorienting—on a diverse set of common objects *only* from human demonstrations in an end-to-end fashion for full robot arms. ***Our key insight is that while human hand-object motion cannot be directly imitated on robots, kinematically retargeted trajectories offer a natural reset distribution of states from which robots can easily explore and access success signals during RL training.*** Drawing inspiration from large-scale RL methods that integrate expert demonstrations into RL training [17, 38, 58], we recognize that object states and kinematically retargeted hand poses are a good starting point to reduce the exploration burden of RL training. Exposure to high-value states automatically trickles down and enables learning from hand-object configurations completely unseen in human demonstrations.

We propose X-RESET, a framework to train dexterous manipulation policies by defining reset distributions over human hand-object states during RL training. X-RESET starts by kinematically retargeting hand-object trajectories to the target embodiment. It then performs object-centric RL in simulation to learn to approach, grasp, and reorient objects to demonstrated goal poses. Key to efficient and scalable training, X-RESET randomly samples hand-object states to supplement exploration from a default home configuration. We additionally propose gravity randomization when resetting mid-trajectory, and find that occasional low-gravity resets help decouple correcting hand-object contact and counteracting external forces. X-RESET can successfully learn from human demonstrations that cover a diverse set of household objects and strategies, and can be used to train different embodiments with a common reward function. Our contributions are summarized as follows:

1) We propose X-RESET, a framework to scale dexterous manipulation skills by framing human hand-object states as a reset distribution for object-centric RL in simulation.
2) We show that X-RESET can learn to approach, grasp, and reorient a wide range of object shapes and sizes from DexYCB [6] with a simple task-agnostic reward function simply by leveraging diverse resets.
3) We evaluate X-RESET along multiple axes, and show that it outperforms prior retargeting baselines in *Task Progress*, exhibits desirable scaling and generalization properties with access to more demonstrations, and is capable of training various embodiments with no reward tuning.

II. RELATED WORK

Imitation Learning via Teleoperation. Imitation learning is the default paradigm for training visuomotor manipulation policies [10, 74, 29, 4], and scales predictably with dataset size [34]. However, these methods require labor-intensive teleoperation of the target robot. A multitude of systems have been developed specifically for this purpose [7, 63, 2], including

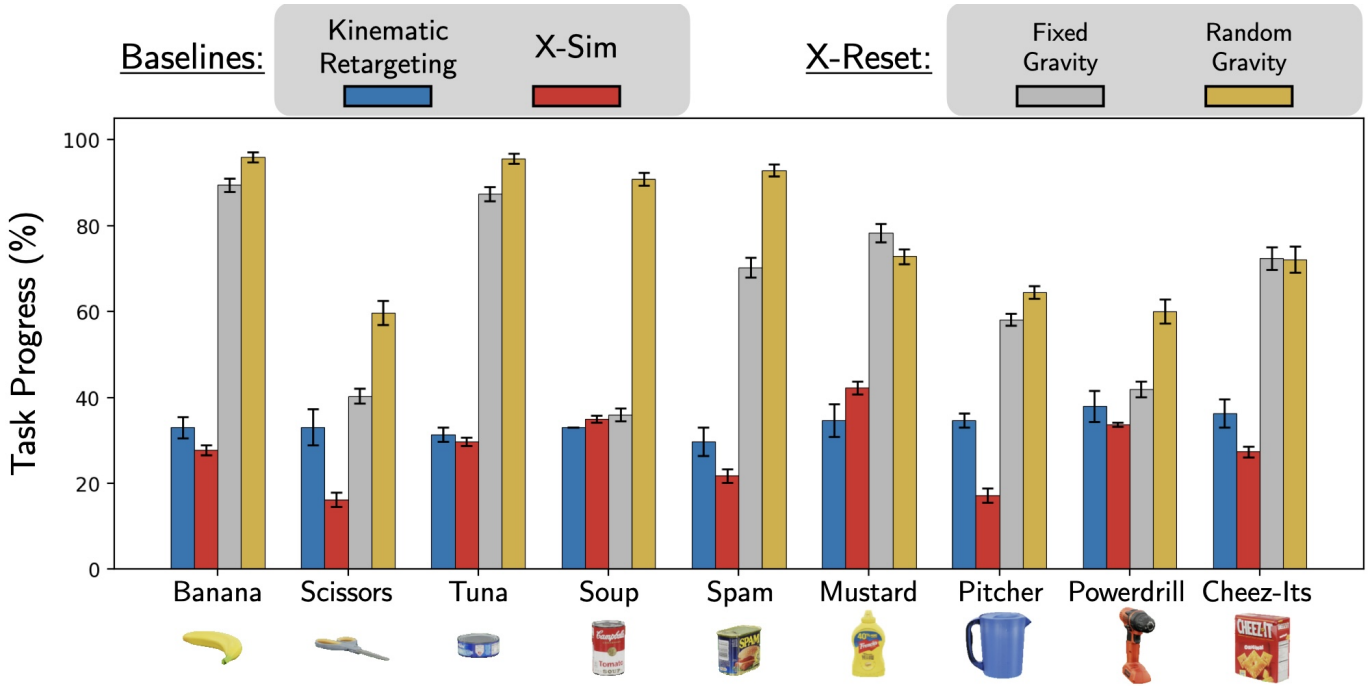

Fig. 3: **Performance on Diverse Objects:** We report *Task Progress (%)* on policies trained to consume 20 target-object demonstrations and evaluate their ability to learn to manipulate objects to goal poses. X-RESET outperforms prior retargeting baselines across all 9 objects, and on average gains a boost in performance when randomizing gravity during hand-object resets.

hardware-specific interfaces [74, 64, 25, 11, 69, 22]. This bottleneck is especially severe for multi-fingered dexterous hands, where high action dimensionality and morphological mismatch with human hands often require thousands of demonstrations to train robust policies [75]. In contrast, human videos are abundant, diverse, and naturally aligned with anthropomorphic hardware—and our approach learns dexterous skills entirely from them, with no robot demonstrations.

Imitating Human Hand Motion. Human videos offer a scalable source of object-interaction priors [18, 12, 24, 35, 6, 16], but lack the action labels needed for standard IL. To bridge this, prior work maps human motion to robot actions through kinematic retargeting [48, 54, 57, 33, 49], narrows the visual gap by overlaying rendered robot arms on human frames [31, 32, 30] or unifying both into a shared keypoint representation [50, 21], or decouples video-derived planning from low-level control [33, 62, 55, 26]. Recent cross-embodiment methods like X-Diffusion [41] instead treat human actions as noised counterparts of robot actions within a diffusion process. These methods assume retargeted hand motions can be directly imitated on the robot, but this often fails due to embodiment mismatch and the true contact dynamics of robot-object interaction [14], and typically require co-training with substantial robot data to recover.

Bridging the Embodiment Gap with RL. Large-scale RL in simulation has produced dexterous skills that transfer to the real world [40, 23, 46, 8, 68] via standard sim-to-real machinery [59, 44, 36], and a complementary direction scales along the object and task axis [67, 61, 1, 3, 27] through hand-engineered priors that remain tractable only for narrow object classes. Closer to our setting, HuDOR [20], X-Sim [14], and Human2Sim2Robot [37] each derive object-centric rewards from human videos and learn robot actions via RL, but use additional modules to work around the large exploration burden for training high-DoF robots from scratch; DexMachina [73] and Dexplore [66] address this with kinematic imitation rewards that require careful per-robot tuning and force strict imitation, forgoing the ability to learn even minor variations of strategies. Collectively, each method covers only a slice of the manipulation arc, and none learns approaching, grasping, and reorienting end-to-end across diverse objects for full robot arms.

We draw inspiration from work that eases RL exploration by resetting to states from expert demonstrations [17, 52, 60, 38, 58]. Retargeted human demonstrations cannot serve directly as expert demos due to kinematic and dynamic inconsistency, but our key insight is that they are nonetheless noisy counterparts of robot states—already spanning the full manipulation arc—and so make a natural reset distribution for object-centric RL. Concurrent work OmniReset [70] also exploits favorable resets but hand-engineers the distribution for low-DoF robots and does not scale to dexterous hands. Inspired by prior in-hand reorientation work [56, 8] that initializes learning with gravity off, we randomize gravity during hand-object resets to let the policy correct kinematic retargeting errors before facing external forces.

III. PROBLEM

We focus on the problem of learning the core dexterous manipulation skills—approaching, grasping, and reorienting

objects to goal poses—purely from human demonstrations. For a given object η, a human hand-object trajectory $\xi^\eta = \{(h_t, s_t)\}_{t=1}^T$ consists of T densely tracked timesteps of MANO [51] hand poses where $h_t \in \mathbb{R}^{K\times 3}$ represents the 3D position of each of the $K = 21$ hand keypoints and $s_t \in \mathrm{SE}(3)$ represents the position and rotation of the object at timestep t. We define the goal pose $g = s_T$ as the last object pose from the human demonstration. We consume a dataset of N human hand-object trajectories $\mathcal{D}^\eta = \{\xi_i^\eta\}_{i=1}^N$, and learn a policy to produce actions $\mathbf{a_t} = \pi_\theta(q_t, o_t, g)$ given robot proprioception q_t, object descriptor o_t (i.e. object pose, point cloud), and goal pose $g \in \mathrm{SE}(3)$. Note that this general formulation allows for jointly training on a dataset of many objects $\mathcal{D} = \bigcup_\eta \mathcal{D}^\eta$.

IV. Approach

We propose X-Reset, a framework for efficiently learning dexterous manipulation skills by resetting the simulator to hand-object states demonstrated by humans during RL training. Our approach consists of two primary stages: (1) generating robot reset states from human hand-object poses; (2) training RL policies in simulation with object-centric rewards by incorporating random resets to demonstrated states.

A. Generating Reset States from Human Hand-Object Demonstrations

We first generate reset states for the robot from human demonstrations. Such states must consist of (a) joint configurations for the robot, and (b) 6D object pose so that they can be spawned in a simulator [39].

Hand-Object Data Processing. We employ an optimization-based kinematic retargeting pipeline (using PyRoki [28]) to estimate robot joint angles corresponding to each frame of the human demonstration. We first transfer human hand pose $h_t \in \mathbb{R}^{K\times 3}$ to robot hand joint angles θ_t^{hand} based on approximate fingertip keypoint targets which varies based on embodiment. Then, we perform collision-aware inverse kinematics (IK) to solve for the remaining arm joints θ_t^{arm} by tracking the wrist over time while keeping the hand poses fixed to finally obtain $\theta_t \in \mathbb{R}^J$ where J is the number of joints in the target embodiment. This process allows us to generate a dataset of L hand-object pairs for the robot $\mathcal{D}^\eta_{\text{robot}} = \{(\theta_l, s_l)\}_{l=1}^L$.

Practical Implementation. In practice, even contact-aware kinematic retargeting from human to robot hand can result in unrealistic states depending on the embodiment gap. To avoid infeasible hand-object simulation resets in the following section, we perform offline filtering to mark and discard unstable hand-object states. These may be a byproduct of (i) inability to solve for robot joint configurations within joint limits or (ii) instability when spawned in simulation (e.g. due to hand-object penetration). See Appendix for more details.

B. Supplementing Object-Centric Reinforcement Learning with Simulation Resets

Next, we perform RL training with a generic object-centric reward function train robots to manipulate objects to goal poses from human demonstrations. Key to our approach is the clever usage of simulation resets: over the course of training, we occasionally reset to hand-object states generated in Section IV-A. By exposing the policy to diverse resets that lie in high-value regions, we are able to propagate success signals from later stages of the object manipulation process and avoid the collapse to local optima often faced by naive exploration from scratch.

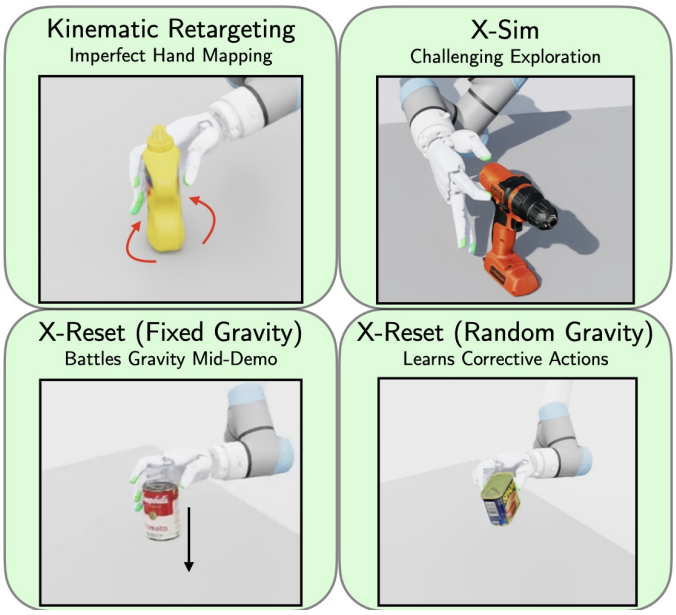

Fig. 4: **Baseline Limitations:** Kinematic Retargeting suffers from improper hand-object and X-Sim collapses due to narrow exploration. X-Reset solves these problems and the use of random gravity helps to learn how to correct kinematic errors in hand-object resets.

Object-Centric Reward Function. We adopt a standard object-centric reward function for training RL policies which is shared across all objects:

$$r(o_t, a_t) = r_{\text{approach}}(o_t) + r_{\text{grasp}}(o_t) + \mathbb{1}_{\text{grasped}} r_{\text{reorient}}(o_t) + r_{\text{success}}(o_t) + r_{\text{smooth}}(o_t, a_t) \tag{1}$$

The first three terms directly correspond to the core skills that underlie dexterous manipulation, incentivizing the robot hand to (i) approach, (ii) grasp, and (iii) reorient objects to their goal pose. r_{success} is a simple bonus for achieving the goal pose (within a ϵ threshold), and r_{smooth} encourages smooth actions. In practice there are various metrics to measure distance between object pose $o_t \in \mathrm{SE}(3)$ and goal pose $g \in \mathrm{SE}(3)$ in order to compute r_{reorient}. We follow [42, 15] and compute the reward as the sum of dense position and rotation rewards:

$$r_{\text{reorient}} = \alpha_{\text{pos}}[1 - \tanh(d_{\text{pos}}(o_t^{\text{pos}}, g^{\text{pos}})] + \alpha_{\text{rot}}[1 - \tanh(d_{\text{rot}}(o_t^{\text{rot}}, g^{\text{rot}})] \tag{2}$$

Additional details about implementation can be found in the Appendix.

Resetting to Hand-Object States. We define a default reset distribution where the robot is in a range of start poses and object is at the start pose of a human demonstration, and randomly choose between resetting to a home configuration *or* a hand-object state generated in Section IV-A. Formally, the

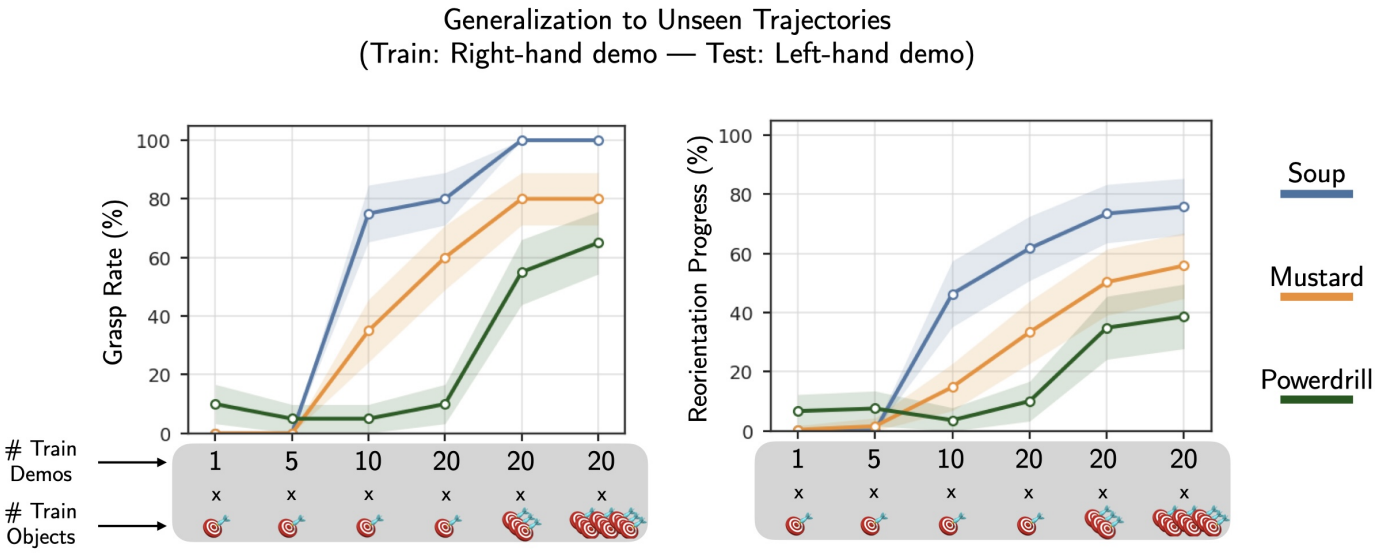

Fig. 5: **Scaling Laws:** We plot *Grasp Rate (%)* and *Reorientation Progress (%)* for policies trained with access to varying numbers of demonstrations (1, 5, 10, 20) and number of objects (1, 3, 9), and find X-RESET shows consistent trends with scale on the three reported tasks.

default reset distribution is defined as ρ_{default}, and ρ_{demo} is a uniform distribution over $\mathcal{D}^{\eta}_{\text{robot}}$ which includes mid-trajectory hand-object states. We sample resets from ρ_{default} with probability $1-p$ and sample from ρ_{demo} with probability p. This reset scheme helps expose the policy π to a diverse set of states in simulation which lie along noisy robot trajectories and can enable discovery of success signals otherwise inaccessible via naive exploration from ρ_{default}.

Gravity Randomization. Randomly sampling resets from $\mathcal{D}^{\eta}_{\text{robot}}$ can have mixed results depending on various factors such as object geometry and contact strategy, and it is common for poor kinematic retargeting outputs to lead to immediate slippage of objects (Fig. 4, bottom left). Instead, we choose to also randomize gravity uniformly between $[-9.81, 0]$ m/s^2 when spawning simulation environments from $\mathcal{D}^{\eta}_{\text{robot}}$. We find that occasional low-gravity resets help decouple correcting hand-object contact and counteracting external forces. Since the policy network is common regardless of gravity, robust strategies that are learned for low-gravity are naturally transferred even for full-gravity resets from ρ_{default}.

Policy Details. We train our policy using PPO [53]. Our policy consumes an object descriptor consisting of 6D object pose (same as offline human demonstrations) and point cloud, a 6D goal pose, and robot proprioception. We make use of an LSTM policy architecture which consumes a short interaction history and can learn adaptive behaviors. We also train our PPO agent with an asymmetric critic [44] that has access to additional privileged information to further stabilize training.

Additional Augmentations. Our method is complementary to existing sim-to-real machinery such as domain randomization [59, 40, 23, 46, 8, 68], and is compatible with other benefits of simulation. In practice, we choose to apply the following augmentations when training our RL policies: (i) expand state coverage with perturbations to robot joint angles and initial object poses from human demos when sampling from ρ_{default}; (ii) observation noise in robot proprioception, object point cloud and pose; (iii) domain randomization over object and robot physics parameters such as mass and friction. See Appendix for more details.

V. EXPERIMENTS

Experimental Setup. We conduct experiments on a 6DoF UR7e arm equipped with a dexterous 22DoF Sharpa right hand, and run ablations with a 7DoF OpenArm with a 1DoF parallel jaw gripper to demonstrate generalization across embodiments with varying morphological gaps from the human. We use hand-object trajectories from DexYCB [6] and evaluate on a representative subset of 9 objects spanning diverse shapes and sizes (Fig. 3). We use IsaacSim [39] for physics simulation. X-RESET (FIXED-G) and X-RESET (RANDOM-G) both use $p = 0.5$ for sampling demo resets and differ only in whether gravity is fixed or randomized during demo resets.

Evaluation Metrics. We train on 20 right-handed demonstrations per object and evaluate on (a) perturbed versions of training configurations and (b) a held-out set of 20 left-handed demonstrations per object—the latter being particularly challenging due to embodiment gap. Our primary metric is *Task Progress (%)*, which measures progress across three equal-weight stages: approaching (≥ 1 finger within $5cm$), grasping (≥ 2 finger contact + lifted $\geq 1cm$), and reorienting (within $2cm$ and 15° of goal). Additional metrics include *Grasp Rate (%)* and *Reorientation Progress (%)*, see Appendix for details on metric computation.

1) **Cross-Embodiment Resets:** How does X-RESET's use of hand-object resets and gravity randomization compare against prior retargeting baselines when learning dexterity from a set of human demonstrations?

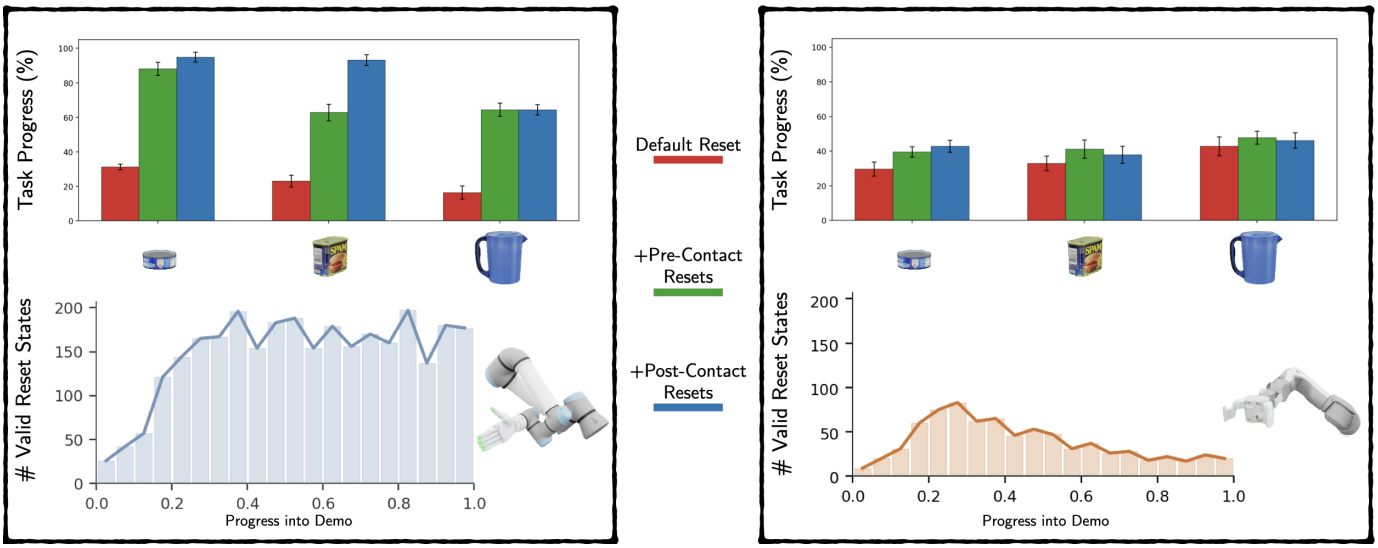

Fig. 6: **Systematic Control of Resets:** We analyze performance of RL training under limited access to hand-object resets. Even when only pre-contact resets are reliably available for the robot, it still supplements training from scratch and massively boosts performance for the UR7e+Sharpa 28DoF robot and more minorly for the OpenArm 8DoF robot with larger embodiment gap.

2) **Scaling Laws:** What are X-RESET's scaling and generalization properties when given access to large-scale demonstrations?
3) **Practical Considerations:** Is X-RESET capable of learning from human demonstrations across different robot hands, and does it still work when accurate hand-object tracking is limited?

A. Cross-Embodiment Resets

We evaluate X-RESET's ability to functionally retarget human demonstrations—approach, grasp, and reorient objects to goal poses purely from human hand-object demonstrations. We compare against two primary baselines which cover the standard approaches for learning from human demonstrations:

- **Kinematic Retargeting:** Used as the foundational backbone for prior works that perform Imitation Learning (IL) from human hand actions [32, 21, 50], this baseline defines an approximate kinematic mapping between 3D MANO hand positions [51] and robot joints. Robot arm and hand joint angles are then solved for via Inverse Kinematics (IK) (Section IV-A), and rolled out *open-loop*.
- **X-Sim [14]:** Extracts object motion from human demonstrations and uses it as a reward function. This baseline ignores all human hand actions, and purely relies on the same generic reward function outlined in Section IV-B to sequentially learn to approach, grasp, and reorient objects.

We evaluate all methods on their functional retargeting capabilities on each of the 20 right-handed human demonstrations per object, and test the closed-loop methods against 5 random perturbations to the initial robot and object poses from ρ_{default}. (Fig. 3). **Kinematic Retargeting** only reliably completes the approach stage of the task, but fails to establish stable grasps (Fig. 4) and consequently rarely progresses further on the majority of objects, hovering near 33% task progress. **X-Sim** exhibits mixed results based on its exploration behavior—for Soup, Mustard, and Powerdrill, hand-object contact opens the gate for the reorientation reward and allows to policy to occasionally grasp the object. However, it breaks down in most cases where it cannot find reliable contact for the 28-DoF robot, and policy collapses to optimizing a mix of r_{approach} and r_{smooth}. On the other hand, X-RESET is exposed to high-value states and success signals by occasionally resetting to mid-trajectory states from ρ_{demo}. Our method can be thought of as a supplement to **X-Sim**, and is able to propagate this knowledge and successfully retarget human demonstrations starting from ρ_{default} which spawns the robot in regions unseen in the human demonstration.

Effects of Gravity Randomization. In addition to the cross-embodiment resets that make object-centric RL tractable for X-RESET we compare two versions of our method: FIXED-G, which uses a fixed, full gravity setting; and RANDOM-G, which samples gravity vectors at uniform random when resetting to demonstration states. We find that both settings are highly capable, with X-RESET (FIXED-G) outperforming baselines by an average of over 30% in task progress. X-RESET (RANDOM-G) generally sees improvements across the board, especially for objects such as Soup where objects are susceptible to slipping from kinematically retargeted hand poses (Fig. 4, bottom left) and learning corrective behavior is easier under low-gravity.

B. Scaling Laws

We evaluate X-RESET's scaling properties along two axes: number of demonstrations per object, and number of objects used to train each policies. We design an experiment to train single-object policies on a range of 1 to 20 demonstrations for the target object, as well two multi-object policies which consumes 20 demonstrations from (a) the three target objects (total

60) and (b) all 9 objects (total 180) together during training (Fig. 5). We aim to test X-RESET's generalization capabilities when tasked with imitating left-handed human demonstrations which are inherently challenging for a right-handed robot arm. On *Grasp Rate (%)* and *Reorientation Progress (%)*, it is very clear that access to more demonstrations of the target object improves grasping and pose reorientation on novel configurations, where the $20\text{demo} \times 1\text{obj}$ policy on average outperforms all policies with access to less demos (Fig. 5). Additionally, our multi-object policy trained on all three target objects outperforms the best single-object policies by jointly training on a vast set of demonstrations, increasing grasp rate by 28.3% and reorientation progress by 19.1%. Incorporating non-target object demonstrations into training additionally boosts the policy's capabilities to grasp and reorient objects from unseen configurations.

C. Practical Considerations

Kinematic retargeting faces two practical limitations: embodiment gap and unreliable hand-object tracking under occlusion, especially post-contact. We test X-RESET under *+Pre-Contact Resets*, which assumes no access to hand-object states after first contact (a natural extension of [37]), and compare against the full method across three reset conditions on Tuna, Spam, and Pitcher for both the Sharpa and OpenArm robots (Fig. 6). Even without object-in-hand states, *+Pre-Contact Resets* outperforms *Default Reset* [15] by 48.2% in task progress for Sharpa, though gains are smaller for OpenArm (7.7%). This disparity reflects OpenArm's larger embodiment gap: its simple parallel jaw yields far fewer valid reset states, let alone post-contact reset states, after retargeting (Fig. 6, bottom). Nonetheless, simulation resets consistently supplement object-centric RL across both embodiments.

VI. DISCUSSION

We present X-RESET, a framework to train dexterous manipulation policies by defining reset distributions over human hand-object states during RL training. By resetting robots to hand-object states demonstrated by humans, our method makes object-centric RL training tractable under a task-agnostic reward function which is otherwise challenging to optimize from naive exploration. We show that X-RESET also exhibits desirable scaling properties with access to more demonstrations and objects, and can still supplement RL training even when post-contact hand-object tracking is unavailable. This work naturally extends to training on larger-scale egocentric human datasets [24, 13] and bimanual robots, and can also be used as a data generation pipeline for distillation to RGB-conditioned policies [4, 29, 70, 14], serving as a stepping stone towards expanding robot action data for dexterous manipulation.

VII. LIMITATIONS

X-RESET was developed using the DexYCB dataset [6], which provides accurate multi-view hand-object tracking. Scaling to in-the-wild monocular video remains an open challenge, though recent advances in occlusion-robust hand tracking [43, 45] are complementary to our approach. We evaluate X-RESET in sim-to-sim transfer and test robustness under changes in physics engine but do not currently have evaluation in the real world. We adopt the asymmetric critic strategy used in recent sim2real works [1] so the policy can directly be deployed with realistic inputs, but our policies can also serve as synthetic data generators for distillation to pure vision-based policies [70]. Our kinematic retargeting assumes sufficient hand anthropomorphism, and transferring knowledge to less human-like end-effectors [5, 71] is an open problem. We currently handle only rigid objects in single-object, short-horizon settings; extending to articulated [73, 65] and deformable [72] manipulation, as well as long-horizon tasks with sparser rewards, is promising future work. Finally, despite using a 22-DoF Sharpa hand with tactile-enabled fingertips, X-RESET does not yet leverage tactile signals from human demonstrations [47, 19], which we leave as a future direction.

REFERENCES

[1] Anonymous. SimToolReal: An object-centric policy for zero-shot dexterous tool use. *OpenReview preprint*, 2026. URL https://openreview.net/pdf?id=sRZ1Aov8wp.

[2] Sridhar Pandian Arunachalam, Irmak Guzey, Soumith Chintala, and Lerrel Pinto. Holo-Dex: Teaching dexterity with immersive mixed reality. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2023. URL https://arxiv.org/abs/2210.06463.

[3] Maria Bauza, Jose Enrique Chen, Valentin Dalibard, Nimrod Gileadi, Roland Hafner, Murilo F. Martins, Joss Moore, Rugile Pevceviciute, Antoine Laurens, Dushyant Rao, Martina Zambelli, Martin Riedmiller, Jon Scholz, Konstantinos Bousmalis, Francesco Nori, and Nicolas Heess. DemoStart: Demonstration-led auto-curriculum applied to sim-to-real with multi-fingered robots. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2025. URL https://arxiv.org/abs/2409.06613.

[4] Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: A vision-language-action model with open-world generalization. In *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pages 17–40. PMLR, 2025. URL https://proceedings.mlr.press/v305/black25a.html.

[5] Hanwen Cao, Hao-Shu Fang, Wenhai Liu, and Cewu Lu. Suctionnet-1billion: A large-scale benchmark for suction

grasping. *IEEE Robotics and Automation Letters*, 6(4): 8718–8725, 2021.

[6] Yu-Wei Chao, Wei Yang, Yu Xiang, Pavlo Molchanov, Ankur Handa, Jonathan Tremblay, Yashraj S. Narang, Karl Van Wyk, Umar Iqbal, Stan Birchfield, Jan Kautz, and Dieter Fox. DexYCB: A benchmark for capturing hand grasping of objects. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. URL https://arxiv.org/abs/2104.04631.

[7] Claire Chen, Zhongchun Yu, Hojung Choi, Mark Cutkosky, and Jeannette Bohg. DexForce: Extracting force-informed actions from kinesthetic demonstrations for dexterous manipulation. *arXiv preprint arXiv:2501.10356*, 2025. URL https://arxiv.org/abs/2501.10356.

[8] Tao Chen, Megha Tippur, Siyang Wu, Vikash Kumar, Edward Adelson, and Pulkit Agrawal. Visual dexterity: In-hand reorientation of novel and complex object shapes. *Science Robotics*, 8(84):eadc9244, 2023. URL https://arxiv.org/abs/2211.11744.

[9] Yuanpei Chen, Chen Wang, Yaodong Yang, and C. Karen Liu. Object-centric dexterous manipulation from human motion data. *arXiv preprint arXiv:2411.04005*, 2024. URL https://arxiv.org/abs/2411.04005.

[10] Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Proceedings of Robotics: Science and Systems (RSS)*, 2023. URL https://arxiv.org/abs/2303.04137.

[11] Cheng Chi, Zhenjia Xu, Chuer Pan, Eric Cousineau, Benjamin Burchfiel, Siyuan Feng, Russ Tedrake, and Shuran Song. Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots. In *Proceedings of Robotics: Science and Systems (RSS)*, 2024. URL https://arxiv.org/abs/2402.10329.

[12] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Scaling egocentric vision: The EPIC-KITCHENS dataset. In *European Conference on Computer Vision (ECCV)*, pages 720–736, 2018. URL https://arxiv.org/abs/1804.02748.

[13] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Jian Ma, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Rescaling egocentric vision: Collection, pipeline and challenges for EPIC-KITCHENS-100. *International Journal of Computer Vision*, 2022. URL https://link.springer.com/article/10.1007/s11263-021-01531-2.

[14] Prithwish Dan, Kushal Kedia, Angela Chao, Edward Duan, Maximus Adrian Pace, Wei-Chiu Ma, and Sanjiban Choudhury. X-Sim: Cross-embodiment learning via real-to-sim-to-real. In *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pages 816–833. PMLR, 2025. URL https://proceedings.mlr.press/v305/dan25a.html.

[15] Prithwish Dan, Kushal Kedia, Angela Chao, Edward Duan, Maximus Adrian Pace, Wei-Chiu Ma, and Sanjiban Choudhury. X-Sim: Cross-embodiment learning via real-to-sim-to-real. In *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pages 816–833. PMLR, 2025. URL https://proceedings.mlr.press/v305/dan25a.html.

[16] Zicong Fan, Omid Taheri, Dimitrios Tzionas, Muhammed Kocabas, Manuel Kaufmann, Michael J. Black, and Otmar Hilliges. ARCTIC: A dataset for dexterous bimanual hand-object manipulation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. URL https://arxiv.org/abs/2204.13662.

[17] Carlos Florensa, David Held, Markus Wulfmeier, Michael Zhang, and Pieter Abbeel. Reverse curriculum generation for reinforcement learning. In *Conference on Robot Learning (CoRL)*, 2017. URL https://arxiv.org/abs/1707.05300.

[18] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4D: Around the world in 3,000 hours of egocentric video. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. URL https://arxiv.org/abs/2110.07058.

[19] Irmak Guzey, Ben Evans, Soumith Chintala, and Lerrel Pinto. Dexterity from touch: Self-supervised pre-training of tactile representations with robotic play. *arXiv preprint arXiv:2303.12076*, 2023.

[20] Irmak Guzey, Yinlong Dai, Georgy Savva, Raunaq Bhirangi, and Lerrel Pinto. Bridging the human to robot dexterity gap through object-oriented rewards. *arXiv preprint arXiv:2410.23289*, 2024. URL https://arxiv.org/abs/2410.23289.

[21] Siddhant Haldar and Lerrel Pinto. Point policy: Unifying observations and actions with key points for robot manipulation. *arXiv preprint arXiv:2502.20391*, 2025. URL https://arxiv.org/abs/2502.20391.

[22] Ankur Handa, Karl Van Wyk, Wei Yang, Jacky Liang, Yu-Wei Chao, Qian Wan, Stan Birchfield, Nathan Ratliff, and Dieter Fox. DexPilot: Vision-based teleoperation of dexterous robotic hand-arm system. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 9164–9170, 2020. URL https://ieeexplore.ieee.org/document/9197124.

[23] Ankur Handa, Arthur Allshire, Viktor Makoviychuk, Aleksei Petrenko, Ritvik Singh, Jingzhou Liu, Denys Makoviichuk, Karl Van Wyk, Alexander Zhurkevich, Balakumar Sundaralingam, Yashraj S. Narang, Jean-Francois Lafleche, Dieter Fox, and Gavriel State. DeXtreme: Transfer of agile in-hand manipulation from simulation to reality. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 5977–5984, 2023. URL https://arxiv.org/abs/2210.13702.

[24] Ryan Hoque, Peide Huang, David J. Yoon, Mouli Siva-

purapu, and Jian Zhang. EgoDex: Learning dexterous manipulation from large-scale egocentric video. *arXiv preprint arXiv:2505.11709*, 2025. URL https://arxiv.org/abs/2505.11709.

[25] Aadhithya Iyer, Zhuoran Peng, Yinlong Dai, Irmak Guzey, Siddhant Haldar, Soumith Chintala, and Lerrel Pinto. OPEN TEACH: A versatile teleoperation system for robotic manipulation. *arXiv preprint arXiv:2403.07870*, 2024. URL https://arxiv.org/abs/2403.07870.

[26] Vidhi Jain, Maria Attarian, Nikhil J. Joshi, Ayzaan Wahid, Danny Driess, Quan Vuong, Pannag R. Sanketi, Pierre Sermanet, Stefan Welker, Christine Chan, Igor Gilitschenski, Yonatan Bisk, and Debidatta Dwibedi. Vid2Robot: End-to-end video-conditioned policy learning with cross-attention transformers. *arXiv preprint arXiv:2403.12943*, 2024. URL https://arxiv.org/abs/2403.12943.

[27] Zhenyu Jiang, Yuqi Xie, Kevin Lin, Zhenjia Xu, Weikang Wan, Ajay Mandlekar, Linxi Fan, and Yuke Zhu. DexMimicGen: Automated data generation for bimanual dexterous manipulation via imitation learning. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2025. URL https://arxiv.org/abs/2410.24185.

[28] Chung Min Kim, Brent Yi, Hongsuk Choi, Yi Ma, Ken Goldberg, and Angjoo Kanazawa. PyRoki: A modular toolkit for robot kinematic optimization. *arXiv preprint arXiv:2505.03728*, 2025. URL https://arxiv.org/abs/2505.03728.

[29] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An open-source vision-language-action model. In *Conference on Robot Learning (CoRL)*, 2024. URL https://arxiv.org/abs/2406.09246.

[30] Marion Lepert, Ria Doshi, and Jeannette Bohg. SHADOW: Leveraging segmentation masks for cross-embodiment policy transfer. In *Conference on Robot Learning (CoRL)*, 2024. URL https://shadow-cross-embodiment.github.io/static/shadow24.pdf.

[31] Marion Lepert, Jiaying Fang, and Jeannette Bohg. PHANTOM: Training robots without robots using only human videos. In *Conference on Robot Learning (CoRL)*, 2025. URL https://arxiv.org/abs/2503.00779.

[32] Marion Lepert, Jiaying Fang, and Jeannette Bohg. Masquerade: Learning from in-the-wild human videos using data-editing. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2026. URL https://arxiv.org/abs/2508.09976.

[33] 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 *Conference on Robot Learning (CoRL)*, 2024. URL https://arxiv.org/abs/2410.11792.

[34] Fanqi Lin, Yingdong Hu, Pingyue Sheng, Chuan Wen, Jiacheng You, and Yang Gao. Data scaling laws in imitation learning for robotic manipulation. *arXiv preprint arXiv:2410.18647*, 2024. URL https://arxiv.org/abs/2410.18647.

[35] Yunze Liu, Yun Liu, Che Jiang, Kangbo Lyu, Weikang Wan, Hao Shen, Boqiang Liang, Zhoujie Fu, He Wang, and Li Yi. HOI4D: A 4D egocentric dataset for category-level human-object interaction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. URL https://arxiv.org/abs/2203.01577.

[36] Tyler Ga Wei Lum, Martin Matak, Viktor Makoviychuk, Ankur Handa, Arthur Allshire, Tucker Hermans, Nathan D. Ratliff, and Karl Van Wyk. DextrAH-G: Pixels-to-action dexterous arm-hand grasping with geometric fabrics. *arXiv preprint arXiv:2407.02274*, 2024. URL https://arxiv.org/abs/2407.02274.

[37] Tyler Ga Wei Lum, Olivia Y. Lee, C. Karen Liu, and Jeannette Bohg. Crossing the human-robot embodiment gap with sim-to-real RL using one human demonstration. *arXiv preprint arXiv:2504.12609*, 2025. URL https://arxiv.org/abs/2504.12609.

[38] Ashvin Nair, Bob McGrew, Marcin Andrychowicz, Wojciech Zaremba, and Pieter Abbeel. Overcoming exploration in reinforcement learning with demonstrations. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 6292–6299, 2018. URL https://arxiv.org/abs/1709.10089.

[39] NVIDIA. NVIDIA Isaac Sim. NVIDIA Developer webpage, 2026. URL https://developer.nvidia.com/isaac/sim/.

[40] OpenAI, Ilge Akkaya, Marcin Andrychowicz, Maciek Chociej, Mateusz Litwin, Bob McGrew, Arthur Petron, Alex Paino, Matthias Plappert, Glenn Powell, Raphael Ribas, Jonas Schneider, Nikolas Tezak, Jerry Tworek, Peter Welinder, Lilian Weng, Qiming Yuan, Wojciech Zaremba, and Lei Zhang. Solving Rubik's cube with a robot hand. *arXiv preprint arXiv:1910.07113*, 2019. URL https://arxiv.org/abs/1910.07113.

[41] Maximus A. Pace, Prithwish Dan, Chuanruo Ning, Atiksh Bhardwaj, Audrey Du, Edward W. Duan, Wei-Chiu Ma, and Kushal Kedia. X-Diffusion: Training diffusion policies on cross-embodiment human demonstrations. *arXiv preprint arXiv:2511.04671*, 2025. URL https://arxiv.org/abs/2511.04671.

[42] Austin Patel, Andrew Wang, Ilija Radosavovic, and Jitendra Malik. Learning to imitate object interactions from internet videos. *arXiv preprint arXiv:2211.13225*, 2022.

[43] Georgios Pavlakos, Dandan Shan, Ilija Radosavovic, Angjoo Kanazawa, David Fouhey, and Jitendra Malik. Reconstructing hands in 3d with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9826–9836, 2024.

[44] Lerrel Pinto, Marcin Andrychowicz, Peter Welinder, Wojciech Zaremba, and Pieter Abbeel. Asymmetric actor critic for image-based robot learning. In *Robotics:*

Science and Systems (RSS), 2018. URL https://arxiv.org/abs/1710.06542.

[45] Rolandos Alexandros Potamias, Jinglei Zhang, Jiankang Deng, and Stefanos Zafeiriou. Wilor: End-to-end 3d hand localization and reconstruction in-the-wild. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 12242–12254, 2025.

[46] Haozhi Qi, Ashish Kumar, Roberto Calandra, Yi Ma, and Jitendra Malik. In-hand object rotation via rapid motor adaptation. In *Proceedings of The 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pages 1722–1732. PMLR, 2023. URL https://proceedings.mlr.press/v205/qi23a.html.

[47] Haozhi Qi, Brent Yi, Sudharshan Suresh, Mike Lambeta, Yi Ma, Roberto Calandra, and Jitendra Malik. General in-hand object rotation with vision and touch. In *Conference on Robot Learning*, pages 2549–2564. PMLR, 2023.

[48] Yuzhe Qin, Yueh-Hua Wu, Shaowei Liu, Hanwen Jiang, Ruihan Yang, Yang Fu, and Xiaolong Wang. DexMV: Imitation learning for dexterous manipulation from human videos. In *European Conference on Computer Vision (ECCV)*, 2022. URL https://arxiv.org/abs/2108.05877.

[49] Ri-Zhao Qiu, Shiqi Yang, Xuxin Cheng, Chaitanya Chawla, Jialong Li, Tairan He, Ge Yan, David J Yoon, Ryan Hoque, Lars Paulsen, et al. Humanoid policy˜ human policy. *arXiv preprint arXiv:2503.13441*, 2025.

[50] Juntao Ren, Priya Sundaresan, Dorsa Sadigh, Sanjiban Choudhury, and Jeannette Bohg. Motion tracks: A unified representation for human-robot transfer in few-shot imitation learning. *arXiv preprint arXiv:2501.06994*, 2025. URL https://arxiv.org/abs/2501.06994.

[51] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics*, 36(6), 2017. URL https://arxiv.org/abs/2201.02610.

[52] Tim Salimans and Richard Chen. Learning Montezuma's revenge from a single demonstration. *arXiv preprint arXiv:1812.03381*, 2018. URL https://arxiv.org/abs/1812.03381.

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

[54] Kenneth Shaw, Shikhar Bahl, and Deepak Pathak. VideoDex: Learning dexterity from internet videos. In *Conference on Robot Learning (CoRL)*, 2022. URL https://arxiv.org/abs/2212.04498.

[55] Junyao Shi, Zhuolun Zhao, Tianyou Wang, Ian Pedroza, Amy Luo, Jie Wang, Jason Ma, and Dinesh Jayaraman. ZeroMimic: Distilling robotic manipulation skills from web videos. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2025. URL https://arxiv.org/abs/2503.23877.

[56] Leon Sievers, Johannes Pitz, and Berthold Bäuml. Learning purely tactile in-hand manipulation with a torque-controlled hand. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2745–2751, 2022. URL https://ieeexplore.ieee.org/document/9811903.

[57] Aravind Sivakumar, Kenneth Shaw, and Deepak Pathak. Robotic telekinesis: Learning a robotic hand imitator by watching humans on YouTube. In *Robotics: Science and Systems (RSS)*, 2022. URL https://arxiv.org/abs/2202.10448.

[58] Stone Tao, Arth Shukla, Tse-kai Chan, and Hao Su. Reverse forward curriculum learning for extreme sample and demo efficiency. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=Q90uzFLjDc.

[59] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 23–30, 2017. URL https://arxiv.org/abs/1703.06907.

[60] Ikechukwu Uchendu, Ted Xiao, Yao Lu, Banghua Zhu, Mengyuan Yan, Josephine Simon, Matthew Bennice, Chuyuan Fu, Cong Ma, Jiantao Jiao, Sergey Levine, and Karol Hausman. Jump-start reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2023. URL https://arxiv.org/abs/2204.02372.

[61] Weikang Wan, Haoran Geng, Yun Liu, Zikang Shan, Yaodong Yang, Li Yi, and He Wang. UniDexGrasp++: Improving dexterous grasping policy learning via geometry-aware curriculum and iterative generalist-specialist learning. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. URL https://arxiv.org/abs/2304.00464.

[62] Chen Wang, Linxi Fan, Jiankai Sun, Ruohan Zhang, Li Fei-Fei, Danfei Xu, Yuke Zhu, and Anima Anandkumar. MimicPlay: Long-horizon imitation learning by watching human play. In *Conference on Robot Learning (CoRL)*, 2023. URL https://arxiv.org/abs/2302.12422.

[63] Chen Wang, Haochen Shi, Weizhuo Wang, Ruohan Zhang, Li Fei-Fei, and C. Karen Liu. DexCap: Scalable and portable mocap data collection system for dexterous manipulation. *arXiv preprint arXiv:2403.07788*, 2024. URL https://arxiv.org/abs/2403.07788.

[64] Philipp Wu, Yide Shentu, Zhongke Yi, Xingyu Lin, and Pieter Abbeel. GELLO: A general, low-cost, and intuitive teleoperation framework for robot manipulators. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024. URL https://arxiv.org/abs/2309.13037.

[65] Hongchi Xia, Entong Su, Marius Memmel, Arhan Jain, Raymond Yu, Numfor Mbiziwo-Tiapo, Ali Farhadi, Abhishek Gupta, Shenlong Wang, and Wei-Chiu Ma. Drawer: Digital reconstruction and articulation with environment realism. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21771–21782, 2025.

[66] Sirui Xu, Yu-Wei Chao, Liuyu Bian, Arsalan Mousavian, Yu-Xiong Wang, Liang-Yan Gui, and Wei Yang.

Dexplore: Scalable neural control for dexterous manipulation from reference-scoped exploration. *arXiv preprint arXiv:2509.09671*, 2025. URL https://arxiv.org/abs/2509.09671.

[67] Yinzhen Xu, Weikang Wan, Jialiang Zhang, Haoran Liu, Zikang Shan, Hao Shen, Ruicheng Wang, Haoran Geng, Yijia Weng, Jiayi Chen, Tengyu Liu, Li Yi, and He Wang. UniDexGrasp: Universal robotic dexterous grasping via learning diverse proposal generation and goal-conditioned policy. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. URL https://arxiv.org/abs/2303.00938.

[68] Max Yang, Chenghua Lu, Alex Church, Yijiong Lin, Chris Ford, Haoran Li, Efi Psomopoulou, David A. W. Barton, and Nathan F. Lepora. AnyRotate: Gravity-invariant in-hand object rotation with sim-to-real touch. *arXiv preprint arXiv:2405.07391*, 2024. URL https://arxiv.org/abs/2405.07391.

[69] Shiqi Yang, Minghuan Liu, Yuzhe Qin, Runyu Ding, Jialong Li, Xuxin Cheng, Ruihan Yang, Sha Yi, and Xiaolong Wang. ACE: A cross-platform visual-exoskeletons system for low-cost dexterous teleoperation. In *Conference on Robot Learning (CoRL)*, 2024. URL https://arxiv.org/abs/2408.11805.

[70] Patrick Yin, Tyler Westenbroek, Zhengyu Zhang, Joshua Tran, Ignacio Dagnino, Eeshani Shilamkar, Numfor Mbiziwo-Tiapo, Simran Bagaria, Xinlei Liu, Galen Mullins, Andrey Kolobov, and Abhishek Gupta. Emergent dexterity via diverse resets and large-scale reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://arxiv.org/abs/2603.15789.

[71] Uksang Yoo, Nathaniel Dennler, Eliot Xing, Maja Matarić, Stefanos Nikolaidis, Jeffrey Ichnowski, and Jean Oh. Soft and compliant contact-rich hair manipulation and care. In *2025 20th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 610–619. IEEE, 2025.

[72] Kaifeng Zhang, Shuo Sha, Hanxiao Jiang, Matthew Loper, Hyunjong Song, Guangyan Cai, Zhuo Xu, Xiaochen Hu, Changxi Zheng, and Yunzhu Li. Real-to-sim robot policy evaluation with gaussian splatting simulation of soft-body interactions. *arXiv preprint arXiv:2511.04665*, 2025.

[73] Mandi Zhao, Yifan Hou, Dieter Fox, Yashraj Narang, Ajay Mandlekar, and Shuran Song. DexMachina: Functional retargeting for bimanual dexterous manipulation. *arXiv preprint arXiv:2505.24853*, 2025. URL https://arxiv.org/abs/2505.24853.

[74] Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Proceedings of Robotics: Science and Systems (RSS)*, 2023. URL https://arxiv.org/abs/2304.13705.

[75] Tony Z. Zhao, Jonathan Tompson, Danny Driess, Pete Florence, Kamyar Ghasemipour, Chelsea Finn, and Ayzaan Wahid. ALOHA unleashed: A simple recipe for robot dexterity. In *Conference on Robot Learning (CoRL)*, 2024. URL https://aloha-unleashed.github.io/.

APPENDIX A
EVALUATION METRICS

Task Progress. We evaluate each rollout with three staged success indicators: approach, grasp, and reorientation. The approach rate measures whether the policy reaches the object-level approach criterion; the grasp rate measures whether sufficient hand-object contact is formed; and the reorientation rate measures final object-pose success.

Task progress is the staged score used in the main figures: 0 for no stage, 0.33 for approach only, 0.66 for grasp only, and 1.0 for successful reorientation.

Grasp Rate. Grasp rate is the percentage of trials where at least the thumb and one other finger make contact with the object and lift it at least 1cm.

Reorientation Progress. For reorientation progress, let $p_t \in \mathbb{R}^3$ and $R_t \in SO(3)$ denote the object pose at time t, and let p_g, R_g denote the goal pose. We define four object-frame keypoints $\mathcal{K} = \{k_i\}_{i=1}^4$, $k_i \in \mathbb{R}^3$, and transform them by the current and goal poses:

$$o_{t,i} = p_t + R_t k_i, \qquad g_i = p_g + R_g k_i.$$

The pose error is the maximum keypoint distance,

$$d(o_t, g) = \max_{i \in \{1,\dots,4\}} \|o_{t,i} - g_i\|_2.$$

We report best-over-episode progress as

$$\text{Progress} = 1 - \frac{\min_t d(o_t, g)}{d(o_0, g)},$$

clipped to $[0, 1]$ for reporting. Tables report the number of evaluated rollouts for each object/method cell when it differs across comparisons.

APPENDIX B
NUMERICAL RESULTS

Tables corresponding to barplots in the main paper are as follows:

- Fig. 3 maps to Tables I, II
- Fig. 5 maps to Tables III, IV, V
- Fig. 6 maps to Tables VI, VII

APPENDIX C
RL TRAINING HYPERPARAMETERS

We use the RL Games library for PPO implementation, and the selected training hyperparameters for Section IV are listed in Table VIII.

Policy observation. The policy receives three observation groups. The policy group contains the object pose, final-goal pose, gravity fraction, and previous action. The proprioception group contains robot joint state and palm/fingertip state (and contact forces for Sharpa Hands, which is obtainable). The perception group is an object point cloud with 64 points and a 5-frame history.

TABLE I: UR7e+SHARPA task-progress comparison across nine objects (Fig. 3)

Object	IK Oracle	X-Sim	Ours (full gravity)	Ours
Banana	33.0 ± 2.4	27.7 ± 1.2	89.5 ± 1.6	95.9 ± 1.1
Scissors	33.1 ± 4.2	16.2 ± 1.7	40.3 ± 1.7	59.7 ± 2.9
Tuna	31.4 ± 1.7	29.7 ± 1.0	87.4 ± 1.6	95.6 ± 1.1
Soup	33.0 ± 0.0	35.0 ± 0.8	36.0 ± 1.4	90.8 ± 1.5
Spam	29.7 ± 3.3	21.8 ± 1.6	70.3 ± 2.3	92.9 ± 1.4
Mustard	34.6 ± 3.8	42.2 ± 1.5	78.3 ± 2.1	72.9 ± 1.7
Pitcher	34.7 ± 1.7	17.2 ± 1.7	58.1 ± 1.4	64.4 ± 1.5
Powerdrill	38.0 ± 3.6	33.7 ± 0.5	42.0 ± 1.8	60.0 ± 2.8
Cheez-Its	36.3 ± 3.3	27.4 ± 1.2	72.4 ± 2.7	72.1 ± 3.0
Avg	33.7	27.9	63.8	78.2

Entries are staged task progress in percent; object rows report mean and standard error.

TABLE II: Macro-averaged stage metrics for the method comparison (Fig. 3)

Method	Task progress	Approach	Grasp	Reorient
IK Oracle	33.7	93.9	7.8	0.6
X-Sim	27.9	80.4	4.0	0.0
Ours (full gravity)	63.8	99.4	64.3	28.7
Ours	78.2	100.0	84.1	51.4

All values are percentages averaged over objects.

TABLE III: Held-out left-hand scaling results for Soup (Fig. 5)

Train demos	Grasp	Keypoint progress	Task progress	Trials
1	0.0	0.2	3.3	20
5	0.0	0.2	0.0	20
10	75.0	46.4	57.8	20
20	80.0	61.7	59.4	20
80	100.0	73.5	67.7	20
180	100.0	75.8	69.4	20

TABLE IV: Held-out left-hand scaling results for Mustard (Fig. 5)

Train demos	Grasp	Keypoint progress	Task progress	Trials
1	0.0	0.4	0.0	20
5	0.0	1.6	11.6	20
10	35.0	14.8	44.5	20
20	60.0	33.4	52.8	20
80	80.0	50.3	59.4	20
180	80.0	55.9	61.1	20

Policy architecture and action. Each point-cloud frame is encoded by a PointNet MLP $3 \to 64 \to 128 \to 64$ with max pooling. The five resulting embeddings are concatenated with the non-point-cloud observations, passed through a 512-unit LSTM with layer normalization, and then through an ELU MLP with layers $[512, 256, 128]$. The policy head outputs the mean of a fixed-log-standard-deviation Gaussian, for example over a 28-dimensional continuous action: 6 relative UR7e joint-position commands and 22 SHARPA implicit-PD hand targets.

Asymmetric critic. The critic receives the same actor observations plus privileged terms: object linear/angular velocity, object pose error to the goal, closest fingertip distance, gravity fraction, and the current reward. These terms are used only for value estimation and are not part of the actor input.

APPENDIX D
IMPLEMENTATION DETAILS

a) DexYCB processing and coordinate frames.: We preprocess DexYCB demonstrations into synchronized hand-object trajectories before retargeting. For each sequence, we extract the MANO hand keypoints, the pose of the grasped YCB object, the object mesh, and approximate hand-object contact points. These quantities are first represented in the native DexYCB camera/world convention.

Before solving IK or launching RL, we map each demonstration into a consistent robot simulation frame. DexYCB provides camera and AprilTag/table calibration, which we use to build a transform $T_{\text{sim}\leftarrow\text{DexYCB}}$ from the recorded camera frame to the simulator table frame. This transform aligns the table normal with the simulator's $+z$ axis, so gravity is upright, and places the object trajectory over the robot workspace. We then apply the same transform to object poses, hand keypoints, contact points, and wrist targets. Finally, we apply a small per-demonstration vertical offset so that the transformed object mesh rests on the simulated table surface. The resulting trajectories are therefore expressed in a single table-aligned frame before kinematic retargeting, IK, reset filtering, and RL training.

b) Contact-aware kinematic retargeting.: We solve MANO-to-SHARPA retargeting as a PyRoki/JAX least-squares problem [28]. For each frame, the decision variables are the SHARPA joint vector q_t and hand root pose T_t. The objective combines local MANO bone-geometry matching, global keypoint alignment, joint/root smoothness, joint-limit costs, and a contact residual from DexYCB object contacts. If $c_{t,j}$ is an observed object contact point and $x_{t,j}(q_t, T_t)$ is the corresponding robot contact-link position, the contact term is

$$E_{\text{contact}} = \sum_{t,j} m_{t,j} \left\| \left[|c_{t,j} - x_{t,j}(q_t, T_t)| - \epsilon \right]_+ \right\|_1,$$

with margin $\epsilon = 1\,\text{cm}$ and weight 5.0. We use weights 10.0 for local alignment, 1.0 for global alignment, and 2.0 for joint/root smoothness.

The retargeted hand root gives a target wrist pose $\hat{T}_t = (\hat{p}_t, \hat{R}_t)$ in calibrated scene coordinates. We then solve a sequential UR7e IK problem with the retargeted hand joints held fixed and only the arm joints active:

$$q_t^a = \arg\min_q \; 50 \left\| p_\ell(q) - \hat{p}_t \right\|_2^2 + 10\, d_R\Big(R_\ell(q), \hat{R}_t\Big)^2 + \lambda_s \left\| q - q_{t-1} \right\|_2^2 + E_{\text{coll}}(q),$$

subject to joint limits. E_{coll} keeps the robot above the table and penalizes self-collision. The first frame is solved with random restarts and later frames are warm-started from q_{t-1}. Frames are valid only when wrist position/orientation error, retargeting error, and arm-joint jumps remain below thresholds.

c) Reset distribution.: Object poses are preserved separately from hand-reset validity, so invalid robot hand states do not remove the demonstration goal trajectory. X-Sim resets place the robot at home with the object at frame 0. X-RESET samples valid retargeted hand-object states from the demonstration; frames whose robot hand state fails IK,

TABLE V: Held-out left-hand scaling results for Powerdrill (Fig. 5)

Train demos	Grasp	Keypoint progress	Task progress	Trials
1	10.0	6.7	36.3	20
5	5.0	7.6	31.4	20
10	5.0	3.7	34.7	20
20	10.0	10.1	38.0	20
80	55.0	34.9	51.2	20
180	65.0	38.7	57.9	20

Values are percentages for Tables III, IV, V. Multi-object rows use 80 demos for four-object training and 180 demos for nine-object training.

TABLE VI: UR7e+Sharpa Reset Strategy Results (Fig. 6)

Method	Tuna	Spam	Pitcher	Macro
Default reset	31.4	23.1	16.5	23.7
+Pre-contact resets	88.1	62.9	64.5	71.8
+Post-contact resets	94.9	93.2	64.4	84.2

retargeting, or reset-validity checks are excluded from the reset distribution. To check for reset-validity, we instantiate each candidate state in Isaac and roll the simulator forward for eight zero-action steps with gravity disabled. States are rejected if they terminate or produce large residual motion, including object linear velocity above $0.10\,\mathrm{m/s}$, object angular velocity above $1.0\,\mathrm{rad/s}$, or robot joint velocity above $2.0\,\mathrm{rad/s}$.

d) Object-centric reward.: The policy is trained with a generic object-centric reward that does not imitate robot joint trajectories. Let C_t be a binary contact gate requiring thumb contact and at least one other finger over the contact threshold. The final-goal tracking terms are

$$r_p = C_t\left(1-\tanh\left(\tfrac{\|p_t-p_g\|_2}{0.2}\right)\right), \qquad r_R = C_t\left(1-\tanh\left(\tfrac{d_R(R_t,R_g)}{1.5}\right)\right),$$

where d_R is quaternion geodesic distance. The total reward is

$$r = r_{\mathrm{reach}} + r_{\mathrm{grasp}} + 4r_p + 4r_R + 10r_{\mathrm{success}} - 0.005\|a_t\|_2^2 - 0.005\|a_t - a_{t-1}\|_2^2 - 500r_{\mathrm{oob}}.$$

Here r_{reach} is a sticky best-so-far fingertip/object tanh reward, r_{grasp} is the two-finger contact bonus, and r_{success} is a contact-gated binary pose success bonus. The two action norm terms incentivize smooth and smaller actions, and r_{oob} is an out-of-bounds penalty that applies if the object leaves the manipulation scene and terminates the episode.

e) Gravity randomization.: The scene gravity remains fixed at deployment gravity $g_0 = -9.81\,\mathrm{m/s^2}$ so the robot always feels the true load. On each reset we sample an object gravity fraction $\alpha \geq 0$ and apply an upward compensation force to the object,

$$F_z = (1-\alpha)m|g_0|,$$

so the effective object gravity is αg_0. Home resets sample near full gravity, $\alpha = 1 + \eta/|g_0|$ with $\eta \sim \mathcal{N}(0,1)$ and clipping. Hand-object resets in the main method sample $\alpha \sim \mathcal{U}(0,1)$; the full-gravity ablation fixes $\alpha = 1$.

TABLE VII: OpenArm Reset Strategy Results (Fig. 6)

Method	Tuna	Spam	Pitcher	Macro
Default reset	29.7	33.0	42.9	35.2
+Pre-contact resets	39.6	41.2	47.9	42.9
+Post-contact resets	42.9	38.0	46.2	42.4

Values are staged task progress percentages for Tables VI, VII.

TABLE VIII: RL training hyperparameters.

Setting	Value
Algorithm	PPO
Discount / GAE	0.99 / 0.95
LR / KL / clip / entropy	1e-3 adaptive / 0.01 / 0.2 / 0.005
Horizon / sequence length / mini-epochs	16 / 4 / 5
Reward scale / grad norm	0.01 / 1.0
Policy input	policy + proprio scalars, followed by perception as the trailing $5 \times 64 \times 3$ object point-cloud history
Point-cloud encoder	Per-frame PointNet MLP $3 \to 64 \to 128 \to 64$ with max pool; concatenate 5 embeddings
Actor architecture	LSTM 512 with layer norm before ELU MLP $[512, 256, 128]$; fixed-log-std Gaussian policy
Action output	28D continuous action: 6 UR7e relative joint-position commands + 22 SHARPA implicit-PD hand targets
Critic input	same actor observations plus object velocity, pose-error, closest-fingertip, gravity, and reward terms
Scale	4 GPU DDP, 4096 envs, minibatch 32768, 10000 epochs

TABLE IX: Domain randomization and reset-time augmentations.

Component	Setting
Object spawn perturbation	$\Delta x, \Delta y \sim \mathcal{U}(-2,2)$ cm; $\Delta\psi \sim \mathcal{U}(-25^\circ, 25^\circ)$
Robot joint perturbation	Home resets only: arm joints $\sigma = 0.075$ rad, hand joints $\sigma = 0.10$ rad
Robot/object friction	Static and dynamic friction $\sim \mathcal{U}(0.5, 1.0)$ with 250 material buckets
Object mass	Mass scale $\sim \mathcal{U}(0.7, 1.3) \times$ default mass values
Hand surface materials	Elastomer friction 0.8; metal friction 0.1
Observation/action interface	64 object points with 5-frame history