Ego-Pi: VLA Fine-Tuning for Ego-Centric Human and Robot Data

Ji Woong Kim[1], Ke Wang[1], Sirui Chen[1], Zipeng Fu[1], Cong Gao[2], Jeff lai[2], Chelsea Finn[1]
[1]Stanford University, [2]Meta
https://egopipaper.github.io/

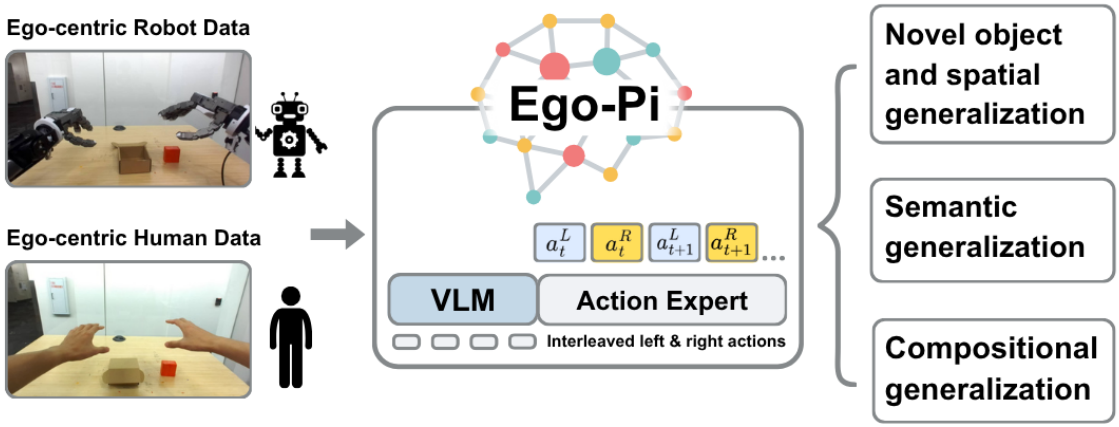


Figure 1. We present Ego-Pi, a vision–language–action framework that co-trains on ego-centric human and robot demonstrations for dexterous humanoid manipulation. Ego-Pi scales VLA models to high-dimensional bimanual actions, allowing robots to generalize across three levels: novel object and spatial generalization, semantic generalization, and compositional generalization.

Abstract

Robotics faces a fundamental challenge of data scarcity. Unlike language or vision research, there is no internet-scale dataset for robotic manipulation. A promising path forward is to leverage ego-centric human data, which can be collected more easily, with greater breadth, and at a larger scale. Towards this end, we investigate key design choices for learning across human and humanoid embodiments equipped with dexterous five-finger hands, using the $\pi_{0.5}$ model as a foundation. We introduce visual guiding keypoints to reduce the embodiment gap between human hands and robot hands, align both domains through a shared action representation, and interleave left and right hand actions across paired tokens to handle high-dimensional hand control without modifying the pre-trained weights. Our results show that human data not only improves robotic generalization but also enables robots to learn new semantics and compose existing skills into novel behaviors without corresponding robot data.

1. Introduction

Robotics faces a data scarcity problem. Unlike text or images, there is no existing internet-scale dataset of robot interactions because it requires physical hardware operating in the real world. A promising way to overcome this limitation is to leverage human data. This idea is becoming increasingly viable as robots become more anthropomorphic,

equipped with arms and dexterous hands that resemble human morphology. In parallel, the rise of ego-centric devices such as the Apple Vision Pro and Meta Ray-Ban glasses, of which 2 million units sold last year [13], has made large-scale collection of human demonstrations more practical than ever. Together, these developments suggest a new opportunity to scale robot learning by co-training on ego-centric human data.

Towards this goal, we study how to co-train human and robot data effectively using a pre-trained vision–language–action (VLA) model. Our central question is one of *generalization*: through co-training with human and robot data, can dexterous robots learn concepts and skills that are only present in the human data?

Specifically, we examine whether human demonstrations can implicitly teach robots high-level semantics and compositional skill generalization. High-level semantics refers to understanding abstract task concepts, such as recognizing the idea of sorting and applying it to unseen object categories. Compositional generalization involves combining known skills to perform new multi-step tasks. These forms of generalization have not been explored in previous studies on human–robot co-training [6–8, 11, 14].

Towards this end, we introduce Ego-Pi (Fig. 1, a framework for adapting pre-trained vision–language–action models to high-dimensional, ego-centric human and humanoid robot data. Ego-Pi employs a token-interleaving strategy that distributes bimanual actions across two tokens, effectively expanding the model's action capacity without modifying pre-trained weights. Since inverse-kinematics-based retargeting is unreliable for high-degree-of-freedom hands, we propose a robot-centric alignment method that directly maps human hand poses into the robot's joint space. Finally, we augment human and robot images with guiding skeleton overlays that visually link corresponding fingers and joints, providing explicit cues to strengthen cross-domain learning.

In our experiments, Ego-Pi consistently outperforms ablation baselines that remove co-training, skeleton overlays, or VLA pretraining. Ego-Pi shows clear gains in both in-distribution and out-of-distribution settings and exhibits behaviors that never emerge from robot data alone. Most importantly, we find that ego-centric human demonstrations can teach high-level semantics and support multi-step composition. Ego-Pi reaches up to 97% success in semantic generalization in a sorting task, and 86% success in compositional generalization in a packaging task, demonstrating that human data provides effective supervision for learning new task mappings and combining previously learned skills.

2. Related Work

Robotics research has increasingly explored ways to leverage human data to address the limited scale of robot demonstrations. Recent work has used ego-centric devices such as the Aria glasses [3] or the Apple Vision Pro to extract hand kinematics from human videos. These reconstructed trajectories, together with the accompanying visual observations, provide a rich source of human demonstrations that can be co-trained with robot data to improve policy performance and generalization.

Several approaches have investigated this idea across different embodiments. EgoMimic [6] showed that co-training human demonstrations with robot data improves performance, although their robot embodiment uses a simple gripper rather than dexterous hands. PH^2D [11] reported similar benefits on a platform equipped with robot fingers, but their models are trained from scratch and therefore do not exploit the broader generalization capabilities of foundation models. EgoVLA [4] addressed this limitation by fine-tuning a foundation model on ego-centric human data, but they did not conduct real-world experiments and operated with a robot hand that has only six actuated joints.

Another line of work attempts to convert human demonstrations directly into robot demonstrations by masking out the human embodiment and overlaying a rendered robot model, as in Masquerade [7] and Mirage [1]. These pipelines involve many stages and are computationally expensive. The rendered overlays are also not occlusion-aware and can obscure the objects being manipulated. While these approaches demonstrate that human

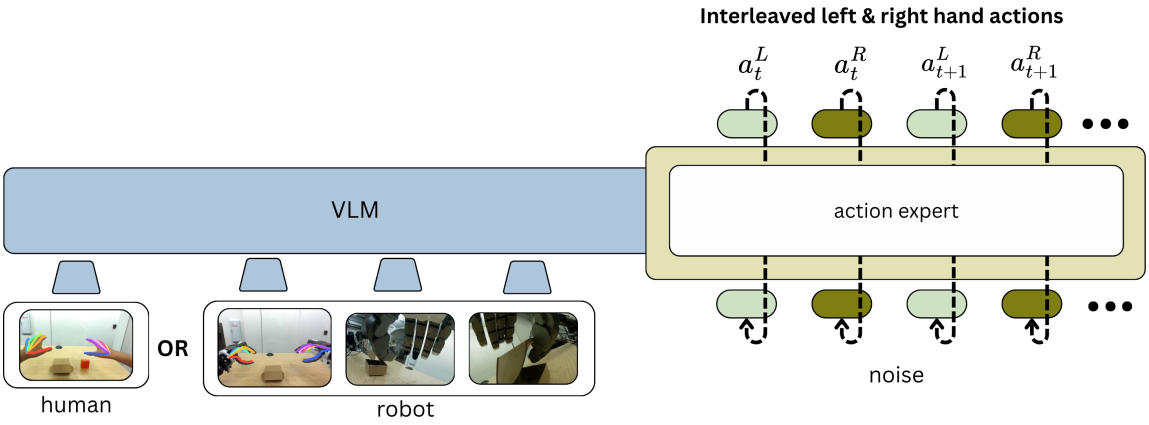


Figure 2. Model architecture of the VLA. The VLM processes either human or robot images, and the action expert produces interleaved actions with two tokens per time step, one for the left hand and one for the right hand. This design accommodates the high-dimensional control of dexterous hands while preserving the original model weights. We use $\pi_{0.5}$ as the base model.

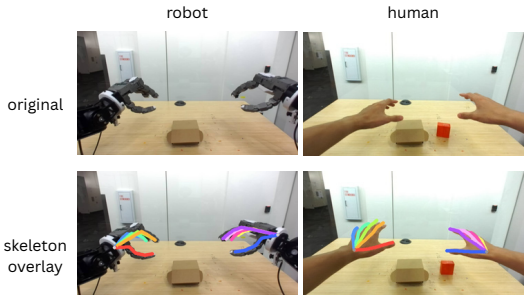


Figure 3. We draw skeleton lines on both the robot and human hands to show how the fingers correspond between them. Each finger is given a unique color that stays the same across both hands. We render the skeleton in a depth-aware, finger-wise manner: fingers that are farther from the camera are drawn behind those that are closer.

data can improve policy robustness in cluttered scenes, none have shown more advanced forms of generalization. In particular, prior work has not demonstrated high-level semantic generalization or multi-step compositional generalization, which are among the central focus of this paper.

3. Ego-Pi

3.1. Preliminaries

We base our model on the flow-matching vision-language-action (VLA) architecture $\pi_{0.5}$ [5], which extends pre-trained vision-language models (VLMs) for robot control. VLAs map multimodal inputs at time t, consisting of a language instruction ℓ_t, images from an ego-centric camera I_t^{ego}, left wrist camera I_t^{L}, and right wrist camera I_t^{R}, together with the proprioceptive state s_t, to short-horizon continuous actions $a_{t:t+H}$. Earlier VLA formulations used autoregressive next-token prediction for actions, but $\pi_{0.5}$ adds a flow-matching action head that enables efficient real-time inference. The model is trained via imitation learning to match expert actions using the following objective:

$$\mathcal{L} = \mathbb{E}_{\tau,\omega}\left[\left|\left|\omega - a_{t:t+H} - f_\theta(a_{t:t+H}^{\tau,\omega}, o_t, \ell_t)\right|\right|^2\right] \tag{1}$$

where $\tau \in [0,1]$ is the flow-time index and $\omega \sim \mathcal{N}(0,I)$ is Gaussian noise. The interpolated action $a_{t:t+H}^{\tau,\omega} = \tau a_{t:t+H} + (1-\tau)\omega$ represents a partially denoised version of the ground-truth action, and $f_\theta(\cdot)$ predicts the corresponding flow velocity field via the action head. The observation tuple is

defined as $o_t = (I_t^{\text{ego}}, I_t^{\text{L}}, I_t^{\text{R}}, s_t)$, which aggregates all sensory inputs.

3.2. Aligning human and robot actions

To effectively transfer behaviors between human and robot embodiments, it is essential to align their policy action representations. In prior works on hand control [2, 4, 11] , a common approach to modeling hand actions involves predicting the wrist position and orientation together with the positions of the five fingertips. For robotic systems, this representation must then be translated into desired joint angles to control the robot hand. This conversion is typically performed through inverse kinematics or an optimization-based method [2, 10] that regresses joint configurations matching the target fingertip positions.

However, for high-dimensional robot hand, such as the one used in our set up (20 actuated joints; see Sec. 3.5), these methods often produce self-colliding or unnatural hand poses. To address this issue, we adopt a robot-centric action representation. Specifically, we first convert human hand keypoints, consisting of 20 keypoints across the hand following the MANO convention [12], into per link joint angles, and then map these angles to the robot joint space. This mapping is necessary because the robot hand differs from the human hand in its kinematic structure and proportions. Details of this mapping process are provided in Supplemental Material.

For the robot hand, we directly use its native joint angles as the action representation, since the human hand configuration has already been mapped to the robot joint space. In summary, the policy hand actions are defined as

$$a = \{\, p,\, r,\, q \,\} \in \mathbb{R}^{29}, \tag{2}$$

where $p \in \mathbb{R}^3$ denotes the wrist position, $r \in \mathbb{R}^6$ represents the wrist orientation in the 6D rotation format [15], and $q \in \mathbb{R}^{20}$ corresponds to the 20 joint angles of the robot hand. The 6D rotation parameterization encodes a rotation matrix $R \in \mathrm{SO}(3)$ using two orthogonal 3D vectors, avoiding discontinuities and singularities inherent in Euler angles or quaternions [15].

3.3. Adapting VLA for hand control

Most vision-language-action (VLA) models, including $\pi_{0.5}$, are trained on robots equipped with parallel-jaw grippers, which have relatively low-dimensional action spaces compared to dexterous robot hands. Specifically, the maximum allowed action dimension of $\pi_{0.5}$ is 32, while our dexterous hand consists of 29 dimensions per hand, totaling 54 dimensions for both hands (Sec. 3.2). The dexterous hand dimensions clearly exceed the 32-dimensional capacity of the original $\pi_{0.5}$ model.

A straightforward approach would be to increase the output dimension of the final projection layer in the $\pi_{0.5}$ action expert. However, in our experiments, this led to high training losses, likely because it disrupted the original pretrained weights.

To avoid modifying the pretrained weights, we propose an interleaved action sequence formulation for bimanual action prediction. To illustrate this, let each action at time t be

$$a_t = \{a_t^L,\, a_t^R\}. \tag{3}$$

The standard formulation predicts an action sequence of horizon H:

$$a_{t:t+H} = \{\, a_t,\, a_{t+1},\, \ldots,\, a_{t+H} \,\}. \tag{4}$$

However, as mentioned above, this approach cannot work because our hand dimensions do not fit within the allowed action dimension of the $\pi_{0.5}$ model. With our proposed interleaving approach, the model emits the left and right actions as separate tokens. In other words, the 54 action dimensions of the dexterous hands are distributed across two 32-dimensional action tokens of the $\pi_{0.5}$ model. Under a fixed action horizon, this allows only $H/2$ bimanual timesteps to be represented:

$$a_{t:t+H/2} = \{\, a_t^L,\, a_t^R,\, \ldots,\, a_{t+H/2}^L,\, a_{t+H/2}^R \,\}. \tag{5}$$

This interleaving approach preserves the pretrained architecture and its parameters. Although it reduces the effective horizon from H to $H/2$, it enables the model to converge well.

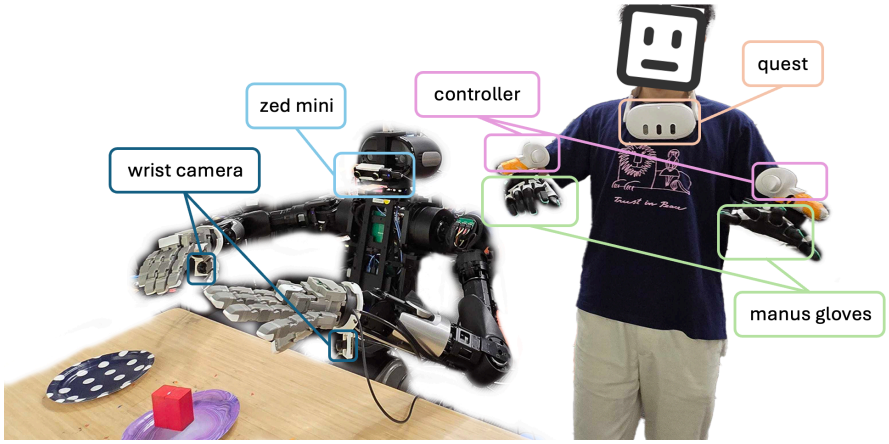


Figure 4. Our experiments use the Galaxea R1 Pro robot equipped with Tesollo dexterous hands (20 joints per hand). The teleoperator controls the robot's wrists and fingers using Quest 3 controllers mounted on Manus gloves. A ZED mini camera and wrist-mounted cameras record the workspace during data collection.

3.4. Visual alignment of human and robot actions

While human and robot hands both have five fingers, their visual appearances differ substantially. To make their visual structure comparable, we overlay guiding skeleton lines on the images before feeding them into the model, as illustrated in Fig. 3. Each finger is assigned a unique color, allowing the model to explicitly understand how fingers from one embodiment correspond to those of the other. The guiding lines are drawn in an occlusion-aware manner so that fingers closer to the camera occlude those farther away. This is achieved by sorting the ground-truth keypoints by their distance from the ego-centric camera, interpolating between them to approximate edge depths, and then plotting the edges from farthest to nearest. This drawing scheme preserves the interpretability of the hand. Without color coding and occlusion-aware plotting, it becomes difficult to discern which lines correspond to which fingers.

3.5. Implementation Details

For the robot setup, we use the Galaxea R1 Pro from Galaxea Dynamics, equipped with a ZED mini mounted on the robot head. Each end-effector is fitted with a Tesollo hand featuring 20 joints per hand. Additionally, Ardurocam wrist cameras are mounted on each wrist, providing a 160° field-of-view.

For teleoperation, we utilize Quest controllers mounted on Manus gloves. The Quest controllers track the 6D pose of the operator's wrists relative to the headset. The robot reproduces the corresponding wrist poses relative to its head using inverse kinematics and PD control. Manus gloves capture the operator's finger joint angles, which are mapped to the robot's hand joints (see Supplemental Material) to enable dexterous control. Both the Manus gloves and Quest controllers record motion data at 100 Hz.

Details of policy training are provided in Supplemental Material. During inference, the policy is deployed on a server cluster using a single NVIDIA H200 GPU. The resulting inference rate is 13 Hz on a single H200 card.

3.6. Data Collection

To collect robot data, the operator wears the Quest 3 on the chest to visualize the robot workspace directly while standing near the robot. The system also supports first-person-view teleoperation,

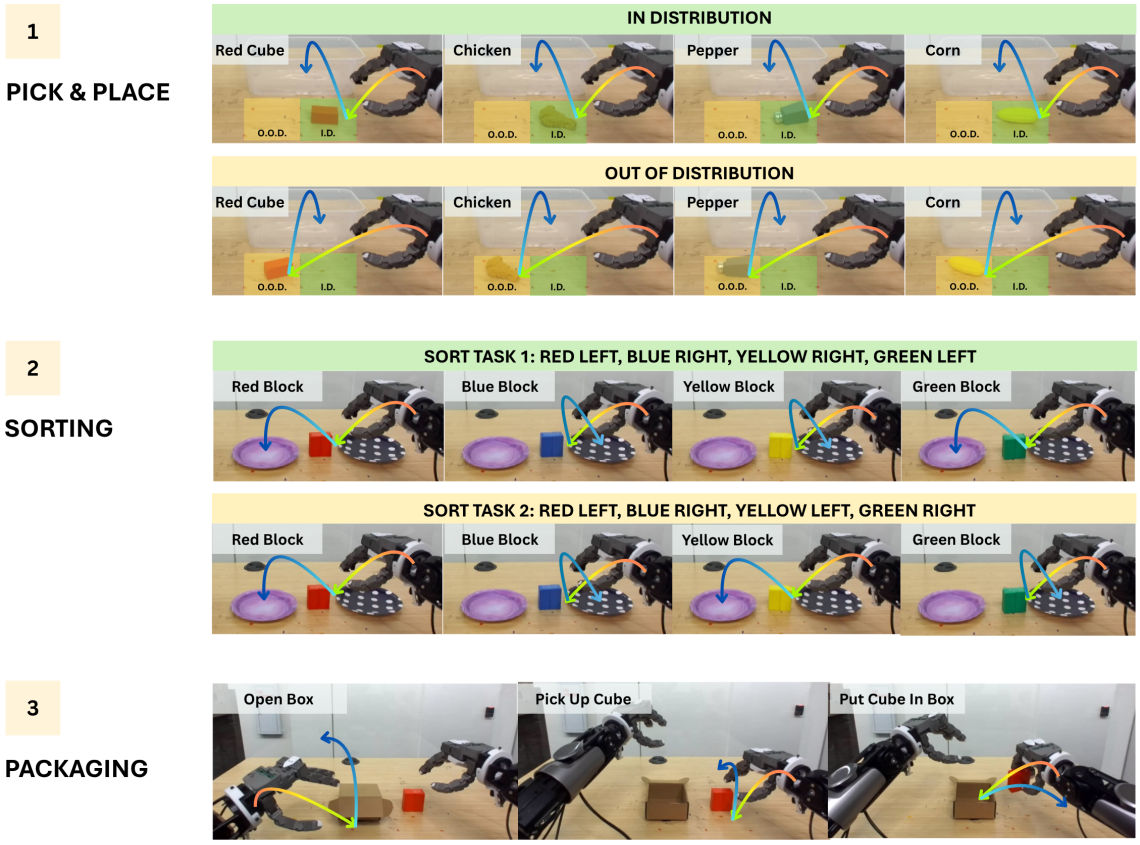


Figure 5. (Top) Pick-and-place task where the robot moves an object into a bin in front. Robot data is collected only within the green region (in-distribution region) but never collected in the yellow region (out-of-distribution region). (Middle) Sorting task where the robot must place each block onto the correct plate based on its color. Robot data is only collected for red and blue blocks, but not yellow and green blocks. (Bottom) Packaging task where the robot opens a box with the left hand and places a block inside.

where the operator wears the Quest 3 headset to view the robot's ego-centric perspective through a ZED mini stereo camera mounted on the robot's head. However, due to the weight of the Quest 3 and discomfort during extended sessions, we primarily used the chest-mounted configuration during data collection.

To collect ego-centric human data, we use a ZED mini stereo camera mounted on a tripod. During data collection, synchronized stereo image pairs are recorded. To extract ground-truth labels for human hand actions, we process these stereo images using HaMeR [9], which reconstructs 21 MANO [12] hand keypoints for each hand. Corresponding keypoints between the stereo views are triangulated to obtain their 3D positions via stereo matching.

To ensure the validity of these reconstructed ground-truth actions, we verify that the replayed trajectories produce physically feasible robot motions in open-loop. In a simple pick-and-place task, we observed that the reconstructed human hand trajectories successfully executed the cube pickup and closely matched the corresponding robot motions,

confirming that the human actions can be reliably transferred to the robot domain in open-loop replay.

4. Experiments

We evaluate how our proposed design choices influence the performance of policies co-trained on human and robot data. Our experiments are organized around three levels of increasing difficulty. We begin by testing whether the policy can perform basic forms of generalization, such as grasping objects that appear only in human data or reaching into spatial regions that were only visited in human data. Prior works have [7, 11] examined such similar level of generalization, so we treat it as a baseline to confirm that the policy acquires the expected capabilities before moving to more challenging settings.

We then investigate whether co-training enables learning high-level semantics and performing multi-step compositions. To this end, we aim to answer the following guiding questions:

1. Can human data help the robot grasp novel objects and reach spatial regions that were absent in robot training?
2. Can human data teach high-level semantics? For example, if the robot learns to sort familiar objects, can it extend this concept to novel object categories after observing human demonstrations?
3. Can human data support compositional reasoning? For example, if the robot can already open boxes and perform pick-and-place actions, can it combine these skills into a packaging task after observing human demonstrations?

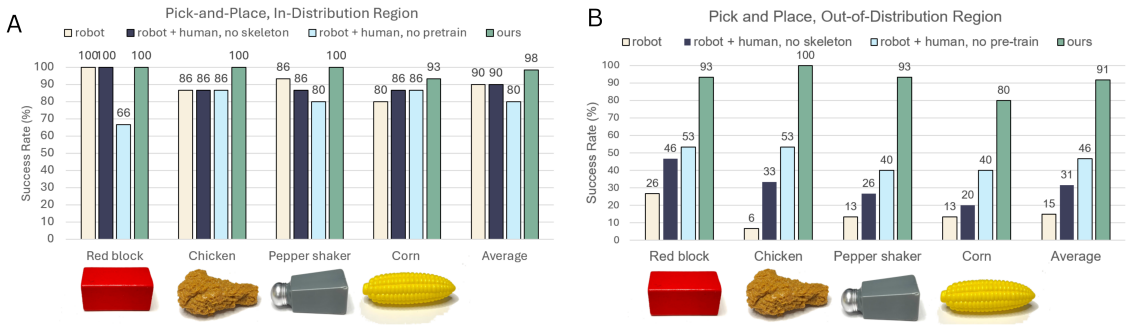
We design three experimental tasks to systematically answer these questions.

Pick-and-place. We design a pick-and-place task where the robot must pick up an object from the table and place it into a bin positioned in front. The goal is to examine whether human-robot co-training enables the robot to reach beyond its training region to perform Pick-and-place of novel objects. To this end, we assign two regions on the table, each measuring 9cm $\times$ 9cm, which we refer to as the in-distribution and out-of-distribution regions, as shown in Fig. 5. In the in-distribution region, we collect 100 teleoperated demonstrations of robot pick-and-place using a red block. No other robot data is collected. We then collect 100 human demonstrations of the same task across both in-distribution and out-of-distribution regions. In addition, we collect 50 human demonstrations for additional objects, including corn, pepper shaker, and a piece of fried chicken in both regions.

Sorting We design a sorting task where the robot must place the correct objects onto two plates positioned in front of it. The goal of this task is to evaluate whether, after co-training on both human and robot data, the robot can correctly sort objects that were only present in human demonstrations. For simplicity, the objects are categorized by color. The robot is trained on a task in which the red block is placed on the left plate and the blue block on the right plate. A total of 85 demonstrations are collected for each of these cases. We then introduce new colored blocks and collect additional demonstrations for these new objects, and only human data is collected. Specifically, we collect human demonstrations in which a yellow block is placed on the left plate and the green block on the right plate, with 100 demonstrations collected for each color.

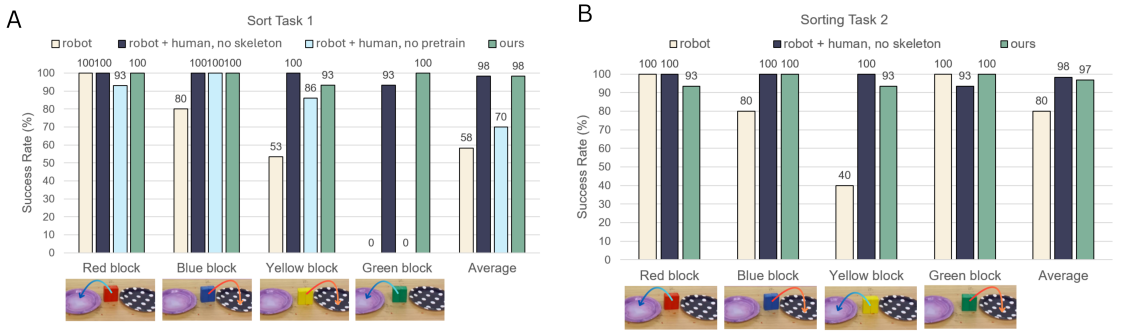
Composition We design a composition task in which the robot must combine previously learned skills to perform a new task. To this end, we first train the robot on two individual skills. The first skill involves opening a box, for which we collect 65 robot demonstrations. The second skill reuses the robot data from the sorting task described above, where the robot performs Pick-and-place actions to move blocks onto plates. At this stage, the robot has learned both how to open a box and how to Pick-and-place objects.

Next, we collect human demonstrations that combine these two skills into a sequential behavior. In this task, the human opens the box and places a block inside it, effectively performing a packaging task. We collect 65 human demonstrations of



Object	In robot data	In human data
Red block	Yes	Yes
Chicken	No	Yes
Pepper shaker	No	Yes
Corn	No	Yes

Figure 6. (A) Pick-and-place performance in the in-distribution setting for both seen and unseen objects. (B) Pick-and-place performance in the out-of-distribution setting for both seen and unseen objects.



Block	Sorting rule	In robot data	In human data
Red	Left plate	Yes	Yes
Blue	Right plate	Yes	Yes
Yellow	Right plate	No	Yes
Green	Left Plate	No	Yes

Block	Sorting rule	In robot data	In human data
Red	Left plate	Yes	Yes
Blue	Right plate	Yes	Yes
Yellow	Left plate	No	Yes
Green	Right plate	No	Yes

Figure 7. (A) Sorting performance where the unseen yellow block is placed on the right plate and the unseen green block on the left plate. (B) Sorting performance in a reversed scenario where the unseen yellow block is placed on the left plate and the unseen green block on the right plate.

this composed sequence. This task tests whether a policy co-trained on human and robot data can perform the packaging task. It examines if the policy can combine previously learned skills based on what it observes from human demonstrations.

Experiment protocol All policies are evaluated 15 times on a given task with a time limit of 30 seconds. If the given policy under evaluation does not complete the task within 30 seconds, the evaluation is marked as failure.

5. Results

Q1: Can human data help the robot pick up novel objects and reach spatial locations the robot has never been to? We test four models under two environments. The four models include 1) a baseline model trained on only robot data, 2) a model co-trained on both human and robot data without skeleton overlay, 3) a model co-trained on both human and robot data with skeleton overlay but without pre-training (i.e. $\pi 0.5$ initialized with original Paligemma weights), 4) and our Ego-Pi model, which includes co-training, skeleton overlay, and fine-tuning from a pre-trained model. The two testing environments include the in-distribution (ID) region (for which we have collected robot pick up data) and out-of-distribution (OOD) region for which we only have human pick-up data. As shown in Fig. 6A, all models were able to pick up all the objects in ID region very well, even including those that were never seen in training data. This is likely due to the use of wrist cameras, which enable the robot to be very competent in grasping objects, even unseen objects.

For Pick-and-place in OOD regions, more clear difference in performance is observed. The model only trained on robot data only averages about 15% success rate. The model co-trained with human data without skeleton overlay shows significant improvement, averaging around 31% success rate. The model without pre-training performs slightly better, averaging at 46% success rate. Our Ego-Pi model performs best at around 90% success rate, largely maintaining its performance across both regions. Qualitatively we observed that our model that performed best had no hesitation when reaching for the OOD region, whereas the other models were not able to reach into the region as though there was an invisible barrier.

Q2: Can human data teach high-level semantics? We similarly test four models: 1) a baseline model trained on robot data only, 2) a model co-trained on human and robot data without skeleton overlays, a 3) model co-trained on human and robot data without pre-training, and 4) our Ego-Pi model. As shown in Fig. 7A, the robot-only baseline is able to sort the seen blocks correctly. However, when introduced with an unseen yellow block (which only appeared in human data), the robot fails to sort this block correctly. Qualitatively we observe that the model is uncertain where to place the yellow block, and sometimes sorts it to the correct plate and sometimes not. When introduced with another unseen object, a green block, the robot is biased to pick a particular plate, thus achieving 0% success rate. Our model performed well, being able to correctly sort the blocks according to how it was shown in the human data. Interestingly, the model co-trained with robot and human data without skeleton overlays performed similarly as the one trained with the skeleton overlays, implying that teaching high-level semantics may not require overlaying skeleton across embodiments to make it explicit.

While we observed promising results, we wanted to absolutely be sure that the co-trained policy was truly attending to the human demonstrations. To demonstrate this, we reversed the sorting semantics of the blocks that appeared in human data. Specifically, instead of placing the yellow blocks to the left plate and the green blocks to the right plate as previously done, we reversed their placement. The same set of models (except the one without pretraining, since this is a redundant experiment) are tested in this revised experiment setting. As shown in Fig. 7B, the models co-trained with human and robot data, both with and without skeleton overlays, were able to follows the reversed sorting semantics by paying attention to the changed human data, aligning with the earlier results. As before, the baseline model (trained only on robot data) is uncertain about how to place the yellow block and achieves a similar success rate as before. For the green block, the success rate is 100% because the baseline model always places the green block on the left plate. Since now the placement semantics are changed (green block must be placed left), the default behavior allows the baseline model to have a very high-success rate.

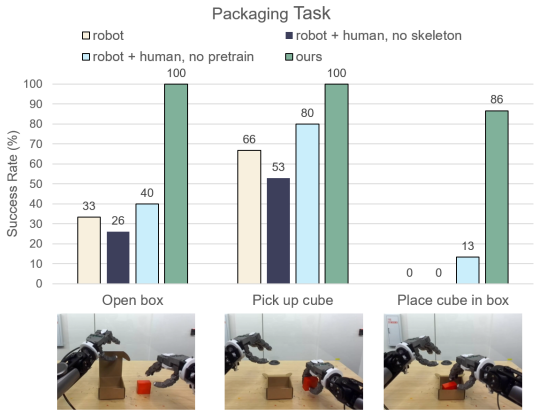


Figure 8. Performance in the packaging task.

Q3: Can human data help with compositional generalization? We similarly test four models, 1) a baseline model trained on robot data only, 2) a model co-trained on human and robot data without skeleton overlay, 3) a co-trained model without pre-training, and 4) our Ego-Pi model. The results are shown in Fig. 8. For the robot-only model, we observe that it often reaches for the box and the cube at the same time, since it lacks the human data showing the sequential relationship of opening the box first and then placing the cube inside. Occasionally, the baseline model successfully opens the box or grabs the cube (sometimes both together), however, it never manages to place the cube inside the box, because it was never taught the cube must go inside the box, which only appeared in human data. Our Ego-Pi model performs quite well, being able to complete the task fully with 86% success rate. Because our model has seen the sequential steps of opening the box then placing, the model would often open the box first while the other hand waits for the box to be fully opened. Only when the box is fully open, the other hand takes a more confident action of placing the block into the box, which is the desirable behavior.

6. Conclusion

We introduced Ego-Pi, a framework that enables effective VLA fine-tuning for ego-centric human and robot data. Ego-Pi expands the action capacity of existing VLA architectures through an interleaved bimanual formulation, aligns human and robot embodiments with a robot-centric action representation, and strengthens visual correspondence through guiding skeleton overlays. Together, these components enable effective co-training on high-dimensional hand actions without modifying the original model weights.

Despite these advantages, our approach has certain limitations. The skeleton overlay is currently overlaid without being occlusion-aware, which can cause the overlaid lines to cover small objects during grasping. This effect is partially mitigated by the use of wrist cameras, which provide an unobstructed view of the object.

References

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

[2] Sirui Chen, Chen Wang, Kaden Nguyen, Li Fei-Fei, and C Karen Liu. Arcap: Collecting high-quality human demonstrations for robot learning with augmented reality feedback. *arXiv preprint arXiv:2410.08464*, 2024.

[3] Jakob Engel, Kiran Somasundaram, Michael Goesele, Albert Sun, Alexander Gamino, Andrew Turner, Arjang Talattof, Arnie Yuan, Bilal Souti, Brighid Meredith, Cheng Peng, Chris Sweeney, Cole Wilson, Dan Barnes, Daniel DeTone, David Caruso, Derek Valleroy, Dinesh Ginjupalli, Duncan Frost, Edward Miller, Elias Mueggler, Evgeniy Oleinik, Fan Zhang, Guruprasad Somasundaram, Gustavo Solaira, Harry Lanaras, Henry Howard-Jenkins, Huixuan Tang, Hyo Jin Kim, Jaime Rivera, Ji Luo, Jing Dong, Julian Straub, Kevin Bailey, Kevin Eckenhoff, Lingni Ma, Luis Pesqueira, Mark Schwesinger, Maurizio Monge, Nan Yang, Nick Charron, Nikhil Raina, Omkar Parkhi, Peter Borschowa, Pierre Moulon, Prince Gupta, Raul Mur-Artal, Robbie Pennington, Sachin Kulkarni, Sagar Miglani, Santosh Gondi, Saransh Solanki, Sean Diener, Shangyi Cheng, Simon Green, Steve Saarinen, Suvam Patra, Tassos Mourikis, Thomas Whelan, Tripti Singh, Vasileios Balntas, Vijay Baiyya, Wilson Dreewes, Xiaqing Pan, Yang

Lou, Yipu Zhao, Yusuf Mansour, Yuyang Zou, Zhaoyang Lv, Zijian Wang, Mingfei Yan, Carl Ren, Renzo De Nardi, and Richard Newcombe. Project aria: A new tool for egocentric multi-modal ai research, 2023.

[4] Ryan Hoque, Peide Huang, David J. Yoon, Mouli Sivapurapu, and Jian Zhang. Egodex: Learning dexterous manipulation from large-scale egocentric video, 2025.

[5] Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael 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, 2025.

[6] Simar Kareer, Dhruv Patel, Ryan Punamiya, Pranay Mathur, Shuo Cheng, Chen Wang, Judy Hoffman, and Danfei Xu. Egomimic: Scaling imitation learning via egocentric video, 2024.

[7] Marion Lepert, Jiaying Fang, and Jeannette Bohg. Masquerade: Learning from in-the-wild human videos using data-editing. *arXiv preprint arXiv:2508.09976*, 2025.

[8] Hao Luo, Yicheng Feng, Wanpeng Zhang, Sipeng Zheng, Ye Wang, Haoqi Yuan, Jiazheng Liu, Chaoyi Xu, Qin Jin, and Zongqing Lu. Being-h0: Vision-language-action pretraining from large-scale human videos, 2025.

[9] Georgios Pavlakos, Dandan Shan, Ilija Radosavovic, Angjoo Kanazawa, David Fouhey, and Jitendra Malik. Reconstructing hands in 3D with transformers. In *CVPR*, 2024.

[10] Yuzhe Qin, Wei Yang, Binghao Huang, Karl Van Wyk, Hao Su, Xiaolong Wang, Yu-Wei Chao, and Dieter Fox. Anyteleop: A general vision-based dexterous robot arm-hand teleoperation system. In *Robotics: Science and Systems*, 2023.

[11] Ri-Zhao Qiu, Shiqi Yang, Xuxin Cheng, Chaitanya Chawla, Jialong Li, Tairan He, Ge Yan, David J. Yoon, Ryan Hoque, Lars Paulsen, Ge Yang, Jian Zhang, Sha Yi, Guanya Shi, and Xiaolong Wang. Humanoid policy human policy, 2025.

[12] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6), 2017.

[13] EssilorLuxottica S.A. Q4 / full year 2024 results – smartglasses as a new driver: Ray-ban meta at 2 million units sold since launch. Press Release, 2025. “Ray-Ban Meta at 2 million units sold since the launch, with strong acceleration in 2024.”.

[14] Ruihan Yang, Qinxi Yu, Yecheng Wu, Rui Yan, Borui Li, An-Chieh Cheng, Xueyan Zou, Yunhao Fang, Xuxin Cheng, Ri-Zhao Qiu, Hongxu Yin, Sifei Liu, Song Han, Yao Lu, and Xiaolong Wang. Egovla: Learning vision-language-action models from egocentric human videos, 2025.

[15] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks, 2020.

Ego-Pi: VLA Fine-Tuning for Ego-Centric Human and Robot Data

Supplementary Material

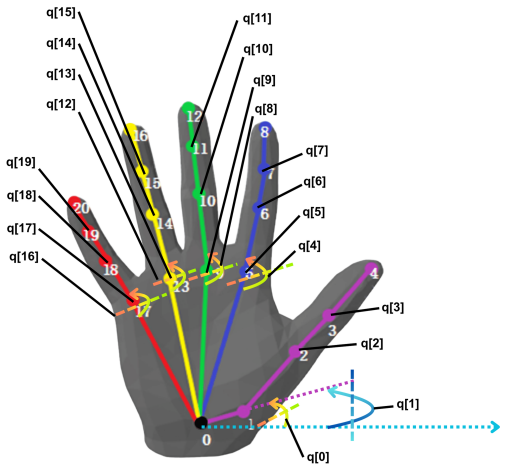


Figure 9. Hand labelled with joint angle locations

7. Joint Mapping between Human and Robot Hands

Let $q \in \mathbb{R}^{20}$ be the human hand joint angles measured by the Manus glove (Fig. 9), and let $q_{\text{robot}} \in \mathbb{R}^{20}$ be the corresponding robot joint angles (e.g., Tesollo hand).

We map each human joint angle to its robot counterpart using a per-joint offset δ_i and a scaling factor f_i:

$$q_{\text{robot},i} = \big(q_i + \delta_i\big)\, f_i, \qquad i \in \{1, \ldots, 20\}. \quad (6)$$

Here, δ_i is an additive offset and f_i is a multiplicative scaling factor. The specific per-joint values used in our implementation are listed below.

```
deg2rad = (pi/180)

q_robot[0] = (38.5 - q[1]) * 0.7 * deg2rad
q_robot[1] = (q[0] + 36.0) * 1.2 * deg2rad
q_robot[2] = (q[2] + 10.0) * 1.2 * deg2rad
q_robot[3] = (q[3] +  5.0) * 1.2 * deg2rad

q_robot[4]  = q[4]  * 1.0 * deg2rad
q_robot[5]  = q[5]  * 1.1 * deg2rad
q_robot[6]  = q[6]  * 1.1 * deg2rad
q_robot[7]  = q[7]  * 1.0 * deg2rad

q_robot[8]  = q[8]  * 1.0 * deg2rad
q_robot[9]  = q[9]  * 1.0 * deg2rad
q_robot[10] = q[10] * 1.0 * deg2rad
q_robot[11] = q[11] * 1.0 * deg2rad

q_robot[12] = q[12] * 1.0 * deg2rad
q_robot[13] = q[13] * 1.0 * deg2rad
q_robot[14] = q[14] * 1.0 * deg2rad
q_robot[15] = q[15] * 1.0 * deg2rad

if q[17] > 55 and q[18] > 25:
    q_robot[16] = abs(q[16]) * 2.0  * 1.0
        * deg2rad
else:
    q_robot[16] = abs(q[16]) / 1.5 * 1.0 *
         deg2rad

q_robot[17] = q[16] * 1.0 * deg2rad
q_robot[18] = q[17] * 1.0 * deg2rad
q_robot[19] = q[19] * 1.0 * deg2rad
```

In order to map human hand joint angles provided by HaMeR to robot hand joints, we perform a similar projection with different offsets and scaling factors, as shown below:

```
q_robot[0]  = q[0]
q_robot[1]  = q[1]
q_robot[2]  = q[2]
q_robot[3]  = q[3]

q_robot[4]  = (q[4] - 0.12) * 1.0
q_robot[5]  = q[5]
q_robot[6]  = (q[6]  + 0.18) * 1.1
q_robot[7]  = (q[7]  + 0.18) * 1.1

q_robot[8]  = q[8]
q_robot[9]  = q[9]
q_robot[10] = (q[10] + 0.18) * 1.1
q_robot[11] = (q[11] + 0.18) * 1.1

q_robot[12] = (q[12] + 0.09) * 1.0
q_robot[13] = q[13]
q_robot[14] = (q[14]) * 0.9
q_robot[15] = q[15]

q_robot[16] = (q[16]) * -1.1
q_robot[17] = (q[17] + 0.24) * 1.0
q_robot[18] = (q[18]) * 1.2
q_robot[19] = (q[19]) * 1.3
```

8. Model Details

The hyperparameters for fine-tuning the $\pi_{0.5}$ policy are shown in in Table 1.

Table 1. Hyperparameters for $\pi_{0.5}$ fine-tuning.

Hyperparameter	Value
Optimizer	AdamW
β_1	0.9
β_2	0.95
Weight Decay	0
Gradient Clip Norm	1.0
LR Schedule	Cosine
Warmup Ratio	0.001
Batch Size	128
Training Steps	5000