HUMAN-TO-ROBOT TRANSFER VIA BRIDGING REPRESENTATION AND ACTION SPACE IN VLAS

Anonymous authors
Paper under double-blind review

ABSTRACT

Pretrained Vision-Language-Action (VLA) policies provide a strong starting point for deployment, but adapting them to a target robot and environment still requires costly embodiment-specific demonstrations. Human demonstrations are far easier to collect, yet directly leveraging them is challenging due to the substantial human-robot embodiment gap, particularly when the underlying VLA has been pretrained primarily on robot data. We introduce **H2RB** (**H**uman-**T**o-**R**obot **B**ridge), a human-robot co-training framework for adapting pretrained VLA policies from unpaired human and robot demonstrations. H2RB addresses the embodiment mismatch at both the representation and action levels: Hand geometry-anchored alignment brings human and robot observations into a compatible task representation, while a decoupled-decoder action tokenizer shares action latents but embodiment-specific action generation. This design allows human demonstrations to provide useful behavioral supervision while retaining the rich visual, semantic, and control priors of the pretrained VLA. Across real-world human-to-robot transfer tasks, H2RB substantially improves object transfer performance from 8.0 to 51.8 and enables skill transfer to score 62.7, in settings where co-train baseline methods totally fail to transfer human-demonstrated skills. The benefit also persists under object-placement shifts, improving out-of-distribution transfer from 6.0 to 45.3. H2RB suggests a way for human demonstrations to serve as a scalable supervision source for adapting pretrained robotic policies, substantially reducing the dependence on costly target-robot data collection.

1 INTRODUCTION

Pretrained robotic policies (*e.g.*, Vision-Language-Action (VLA) models; Brohan et al., 2023; Kim et al., 2024) provide a strong starting point for adapting general robotic capabilities to a target deployment environment. However, a major bottleneck remains in the collection of target robot demonstrations. While these policies typically require a substantial number of expert demonstrations to achieve reliable performance, collecting such data on physical robots is costly, time-consuming, and difficult to scale. It requires access to the physical platform, a data collection pipeline (e.g., teleoperation), and repeated environment resets (Brohan et al., 2022; Khazatsky et al., 2024).

Human demonstrations, in contrast, can be collected quickly and naturally without relying on dedicated teleoperation systems (Chi et al., 2024; Qiu et al., 2025; Yang et al., 2025; Wang et al., 2023), using only a lightweight camera setup. Yet, directly leveraging them to adapt robot policies remains challenging due to the substantial embodiment gap between humans and robots, particularly when the pretrained policy has been trained primarily on robot data.

Prior works bridge human and robot demonstrations by placing them in a common action space, either through latent actions inferred from frame transitions (Ye et al., 2025; Chen et al., 2026; Dai et al., 2026; Bu et al., 2025; Chen et al., 2025) or through reconstructed world-space hand motion (Li et al., 2025; Fu et al., 2026; Hoque et al., 2026). While these approaches reduce the action-space mismatch across embodiments, they leave the perceptual gap largely unresolved. As a result, VLA policies must still map visually and kinematically distinct human and robot observations into the same action representation, making it difficult to reliably translate behaviors learned from human demonstrations into valid robot actions.

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

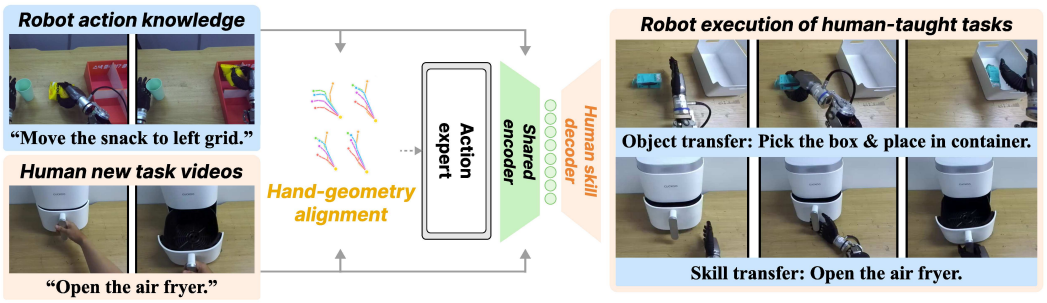


Figure 1: **Human-to-robot task transfer with H2RB.** H2RB transfers task experience from human videos to robots on VLAs by bridging the cross-embodiment perception and action gaps, without target-robot demonstrations or paired demonstrations.

This motivates us to revisit human-to-robot transfer from a different perspective: rather than only translating human actions into a robot-compatible space, we seek to make human demonstrations compatible with a pretrained robot policy at both the representation and action levels. We hypothesize that successful transfer requires two complementary properties. First, human and robot observations at similar task progress should induce compatible perceptual representations despite their embodiment differences. Second, the policy should preserve embodiment-specific action realization rather than forcing heterogeneous behaviors through a single shared mapping.

Our approach. Based on this insight, we introduce H2RB, a training framework that incorporates human demonstrations into pretrained VLA policies without paired human-robot trajectories. H2RB bridges the embodiment gap at both the perceptual and action interfaces. First, a geometry-anchored alignment uses wrist–fingertip geometry to ground VLM output features across embodiments, without paired demonstrations or human-to-robot observation synthesis. Second, a decoupled-decoder latent tokenizer separates shared action representation from embodiment-specific realization: a shared encoder maps human and robot actions into a common latent space, while embodiment-specific decoders recover their respective action distributions. Together, these components enable human-demonstrated behaviors to transfer across embodiments and be executed through the appropriate action interface, as illustrated in Figure 1.

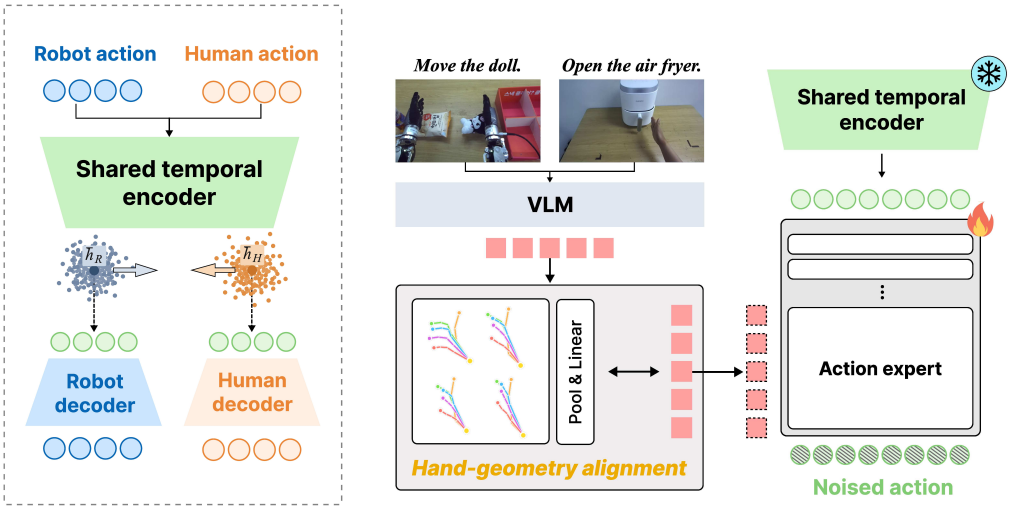
We evaluate H2RB on two complementary forms of human-to-robot adaptation: within-skill object transfer and skill-level transfer. In both settings, target behaviors are provided only through human demonstrations. Compared with matched human-robot co-training and robot-data baselines, H2RB improves real-world object transfer from 8.0 to 51.8 task score and skill transfer to 62.7 where the baselines totally fail. With heterogeneous human-robot large scale training, performance further increases to 56.9 and 84.7, respectively, indicating that H2RB continues to benefit as cross-embodiment training scales.

2 Method

H2RB transfers behaviors from human videos to robot execution by bridging gaps in action representation and observation conditioning. We learn shared latent actions over a common wrist–hand interface with embodiment-specific decoders, and align the action expert's conditioning features across embodiments using hand geometry.

2.1 Preliminaries

At timestep t, a Vision-Language-Action model (VLA) predicts an action chunk $\mathbf{A}_t = [\mathbf{a}_t, \ldots, \mathbf{a}_{t+K-1}]$ from visual observations $\mathbf{O}_t$, a language instruction $\mathbf{c}$, and the current state $\mathbf{s}_t$. A pretrained VLM produces conditioning features $\mathbf{h}_t = F(\mathbf{O}_t, \mathbf{c})$, which an action expert uses with $\mathbf{s}_t$ to generate the action chunk. Following flow-based VLAs (Bjorck et al., 2025; Black et al., 2024; Kim et al., 2026a), we use conditional flow matching (Lipman et al., 2022): for Gaussian noise $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and flow time $\tau \in [0, 1]$, we define $\mathbf{A}_{t,\tau} = (1-\tau)\boldsymbol{\epsilon} + \tau\mathbf{A}_t$ and optimize $\mathcal{L}_{\text{FM}} = \mathbb{E}[\|G_\theta(\mathbf{A}_{t,\tau}, \tau, \mathbf{s}_t, \mathbf{h}_t) - (\mathbf{A}_t - \boldsymbol{\epsilon})\|_F^2]$.



(a) Decoupled-decoder latent tokenizer **(b) Hand geometry-anchored alignment**

Figure 2: **Overview of H2RB. (a)** A shared temporal encoder maps human and robot action chunks into a common latent action space, while embodiment-specific decoders reconstruct actions for each domain. **(b)** Human and robot demonstrations jointly train the VLA action expert. Hand geometry-anchored alignment makes the VLA conditioning representations more compatible across embodiments, while the frozen shared action encoder provides latent action targets.

2.2 Learning a Shared Action Space

To reduce the action gap between human and robot demonstrations, we first express their motions in a common physical parameterization. We then learn a shared latent prediction space while retaining separate decoders for embodiment-dependent trajectory structure.

Shared wrist-hand parameterization. For human videos, we first use a HaWoR-based (Zhang et al., 2025a) reconstruction pipeline to recover wrist motion and hand articulation. The reconstructed motion is then transformed into the robot coordinate convention and retargeted to the robot hand. Human and robot actions are expressed in a common wrist–hand representation (Kim et al., 2026b), $\mathbf{a}_t^d = [\mathbf{p}_t^d, \mathbf{r}_t^d, \mathbf{q}_t^d] \in \mathbb{R}^{9+D_q}$ for $d \in \{H, R\}$, where $\mathbf{p}_t^d$ and $\mathbf{r}_t^d$ denote the 3D wrist position and 6D wrist rotation, respectively, and $\mathbf{q}_t^d \in \mathbb{R}^{D_q}$ denotes the hand configuration. At deployment, the predicted wrist pose is converted into robot arm motion via inverse kinematics.

Latent action divergence on VLAs across embodiments. A shared wrist–hand parameterization alone does not fully bridge the gap. Because the VLA is conditioned on visually distinct human and robot observations, its action expert can produce systematically different latent action distributions even when both are trained to predict the same physical action space. In practice, we observe that human-conditioned and robot-conditioned predictions separate into source-dependent latent modes, and decoding these latents through a shared action mapping tends to show robot-like action realization under robot observations, even for tasks learned through human demonstrations.

Decoupled-decoder latent tokenizer. To account for this source dependence, we learn a shared latent action space while retaining embodiment-specific decoders. As shown in Figure 2a, we train a variational action autoencoder (Kingma & Welling, 2013) with a shared encoder and separate human and robot decoders:

$$\mathbf{Z}_t^d \sim q_\psi(\cdot \mid \mathbf{A}_t^d), \qquad \widehat{\mathbf{A}}_t^d = D_d(\mathbf{Z}_t^d), \quad d \in \{H, R\}. \tag{1}$$

The shared encoder maps human and robot actions into a common latent space, while source-specific decoders model the distinct action realizations of each embodiment. We optimize the tokenizer with reconstruction and KL objectives, and further couple the two latent distributions through position-wise centroid matching with variance and covariance regularization (see Appendix A). This encourages both sources to occupy a compatible latent prediction space without forcing their source-dependent trajectory structures to collapse into a single decoder.

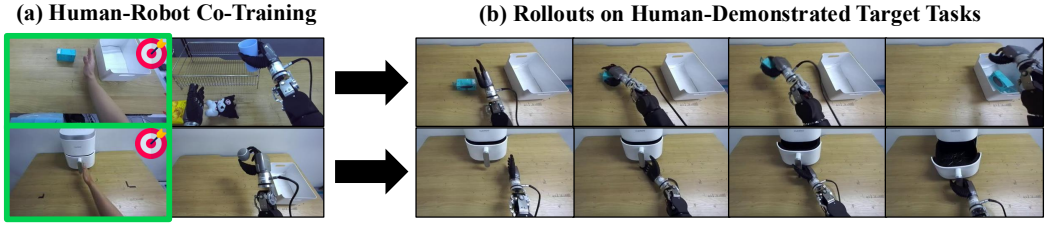


Figure 3: **Qualitative human-to-robot transfer rollouts.** (a) H2RB is trained jointly on human and robot demonstrations. (b) The trained policy executes target tasks introduced through human demonstrations, including object transfer on box, followed by skill transfer on opening.

After training the tokenizer, we freeze both encoder and decoders and use the posterior mean $\mathbf{Z}_t^d = E_\psi(\mathbf{A}_t^d)$ as the policy target. As shown in Figure 2b, the VLA thus predicts latent actions from both human and robot demonstrations under the same flow-matching objective, denoted as $\mathcal{L}_{\text{pred}}$.

2.3 Hand Geometry-Anchored Alignment

Aligning the action space alone does not resolve the substantial visual differences between human and robot observations. We therefore anchor the VLM output features, which serve as conditioning representations for the action expert, to hand geometry that is directly comparable across embodiments.

Geometry-anchored VLM conditioning. We ground the output features from the VLM in the current wrist pose and five wrist-local fingertip positions, denoted by $\mathbf{g}_t$ (see Figure 2b). Hand geometry targets for human demonstrations are obtained from the wrist-hand representation of Section 2.2, and those for robot demonstrations are directly derived from the robot's configurations. A shared residual adapter A_ϕ and readout R_ω produce $\mathbf{h}_t^{\text{geo}} = \mathbf{h}_t + A_\phi(\mathbf{h}_t)$ and $\widehat{\mathbf{g}}_t = R_\omega(\text{Pool}(\mathbf{h}_t^{\text{geo}}))$, trained with a masked SmoothL1 objective $\mathcal{L}_{\text{geo}}$. The geometry readout is used only during training, while the action expert is conditioned directly on the adapted token sequence $\mathbf{h}_t^{\text{geo}}$. Because each observation is supervised by its own geometry, human and robot data can be anchored to the same physical representation without requiring paired demonstrations or temporal alignment.

The action expert is conditioned on $\mathbf{C}_t$ and the current state. Its prediction objective and the geometric auxiliary objective jointly train the policy:

$$\mathcal{L}_{\text{VLA}} = \mathcal{L}_{\text{pred}} + \lambda_{\text{geo}} \mathcal{L}_{\text{geo}}. \tag{2}$$

The VLM backbone, tokenizer, and both reconstruction decoders remain frozen during this policy-training stage. At deployment, robot observations, state, and instruction condition the latent-action policy π_θ, and the generated sequence is decoded through the frozen human branch for human-sourced tasks:

$$\mathbf{Z}_t^{\text{gen}} \sim \pi_\theta(\cdot \mid \mathbf{C}_t, \mathbf{s}_t), \qquad \mathbf{A}_t^{\text{cmd}} = D_H(\mathbf{Z}_t^{\text{gen}}). \tag{3}$$

The decoder output is converted to robot commands using its prescribed inverse normalization and coordinate mapping. Both decoders reconstruct the same robot-compatible wrist-hand representation. Although deployment uses D_H, the shared encoder and latent-action policy are learned from both human and robot demonstrations.

3 Experiments

We evaluate H2RB in two stages. First, under controlled task-specific fine-tuning, we test whether human demonstrations transfer target objects and target skills to a robot beyond vanilla human-robot co-training. Second, we test whether the same cross-embodiment principle remains useful when incorporated into heterogeneous multi-embodiment mid-training.

3.1 Experimental Setup

Our real-world transfer experiments use an OpenArm humanoid with two Inspire hands, with policies initialized from RLDX-1-IMG (Kim et al., 2026a), a robot-only pretrained VLA. Human

Table 1: **Human-to-robot object transfer.** Robot in-domain reports performance on a robot-data-trained task, while Human transfer evaluates target tasks introduced through human demonstrations. Each entry is the weighted task score (0-100).

	Robot in-domain	Human transfer			
Method	Doll	Box	Ball	Towel	Avg.
Base policy human-only	31.5	34.3	26.0	27.0	29.1
Base policy in-domain-only	55.0	18.0	14.0	4.0	12.0
Base policy co-train	18.0	12.0	4.0	8.0	8.0
+ HaWoR-shared action (Zhang et al., 2025a)	10.5	11.0	12.0	9.0	10.7
+ Ego-Pi-shared action (Kim et al., 2026b)	30.0	38.3	17.3	26.0	27.2
+ H2RB (Ours)	**49.0**	**53.0**	**49.0**	**53.3**	**51.8**

Table 2: **Human-to-robot skill transfer.** Robot in-domain reports performance on a robot-data-trained task, while Human transfer evaluates target tasks introduced through human demonstrations. Each entry is the weighted task score (0-100).

	Robot in-domain	Human transfer		
Method	Box PnP	Open	Close	Avg.
Base policy human-only	0.0	36.0	22.0	29.0
Base policy in-domain-only	80.5	0.0	0.0	0.0
Base policy co-train	16.5	0.0	0.0	0.0
+ HaWoR-shared action (Zhang et al., 2025a)	19.5	0.0	0.0	0.0
+ Ego-Pi-shared action (Kim et al., 2026b)	70.5	0.0	0.0	0.0
+ H2RB (Ours)	**81.0**	**50.7**	**74.7**	**62.7**

demonstrations are labeled using the wrist-hand action representation in Section 2.2. Our co-train baseline trains the same VLA on matched human and robot data with separate embodiment-specific action representations, while the HaWoR-shared action (Zhang et al., 2025a) and Ego-Pi-shared action (Kim et al., 2026b) baselines use their respective shared action representations, but without our hand geometry-anchored alignment or decoupled-decoder latent tokenizer. All comparisons use the same adaptation data and training budget, fully described in Appendix B.1.

We study two axes of human-to-robot transfer. *Object transfer* introduces target objects through human pick-and-place demonstrations while retaining prior robot pick-and-place experience. *Skill transfer* introduces opening and closing interactions absent from the source robot demonstrations. (see Figure 3) For each task, we define four ordered progress stages and assign each rollout a score based on the highest stage reached, allowing partial task completion to receive subtask-level weighted task score from 0 to 100. Further details are provided in Appendix B.2.

3.2 Human-to-robot task transfer

Object transfer results. Table 1 summarizes the results on manipulating target objects introduced only through human demonstrations. The robot-only policy performs well on the in-domain Doll PnP task (55.0) but generalizes poorly to unseen target objects (12.0), despite the prior knowledge encoded in the pretrained VLA. Human-only training, in contrast, captures the demonstrated interactions but lacks the precision required for reliable robot execution, particularly near contact. H2RB improves object-transfer performance from 8.0 to 51.8 (+43.8) while preserving in-domain robot performance, whereas the other baselines degrade largely. These results suggest that bridging the human–robot perceptual gap enables human demonstrations to introduce new object interactions without sacrificing the capabilities inherited from the robot-pretrained VLA.

Skill transfer results. Skill transfer exposes a different bottleneck, as shown in Table 2. HaWoR-shared action fails to transfer the Open/Close skills (0.0) and substantially degrades performance on the familiar robot task to 19.5. Ego-Pi-shared action restores robot-compatible execution to 70.5 on the familiar task, yet still achieves 0.0 on both target skills. Its rollouts instead tend to reproduce

Table 3: **Object transfer performance under evaluation object placement distribution shift.** Each target object is evaluated with 20 in-distribution (ID) and 10 out-of-distribution (OOD) placement rollouts. Avg. aggregates the three objects. All values are weighted task scores (0-100).

	ID				OOD			
Method	Box	Ball	Towel	Avg.	Box	Ball	Towel	Avg.
Base policy co-train	13.5	3.0	10.5	9.0	9.0	6.0	3.0	6.0
+ HaWoR-shared action	10.5	10.5	9.0	10.0	12.0	15.0	9.0	12.0
+ Ego-Pi-shared action	45.0	18.5	27.0	30.2	25.0	15.0	24.0	21.3
+ H2RB (Ours)	**60.0**	**50.0**	**55.0**	**55.0**	**39.0**	**47.0**	**50.0**	**45.3**

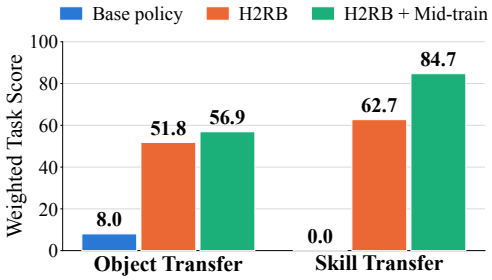


Figure 4: **Effect of human-robot mid-training.** Mid-training improves transfer abilities, indicating the scalability of our method.

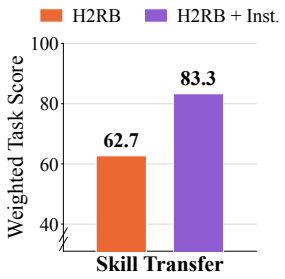


Figure 5: **Effect of instruction conditioning.** Instruction conditioning improves skill transfer abilities.

familiar in-domain behaviors such as Pick-and-Place, indicating that a shared, executable action space alone is insufficient to transfer the task semantics introduced through human demonstrations. By incorporating embodiment-specific decoding, H2RB improves skill-transfer performance from 0.0 to 62.7. This result highlights the importance of separating shared latent skill representation from embodiment-specific action realization when transferring behaviors across different motion distributions. Representative successful rollouts for each transfer task are shown in Figure 9.

Transfer under distribution shift. We further evaluate object transfer when the rollout distribution departs from the adaptation setting. Table 3 reports the score for the in-distribution and out-of-distribution sets. Retargeting human motion into a robot-compatible action space consistently improves over directly using a shared hand-action format. H2RB further improves the score from 30.2 to 55.0 (+24.8) in distribution and from 21.3 to 45.3 (+24.0) out of distribution. The gains indicate that hand geometry-anchored alignment further reduces the perception gap and improves transfer to novel object locations. Detailed breakdown and analysis are provided in Appendix C.

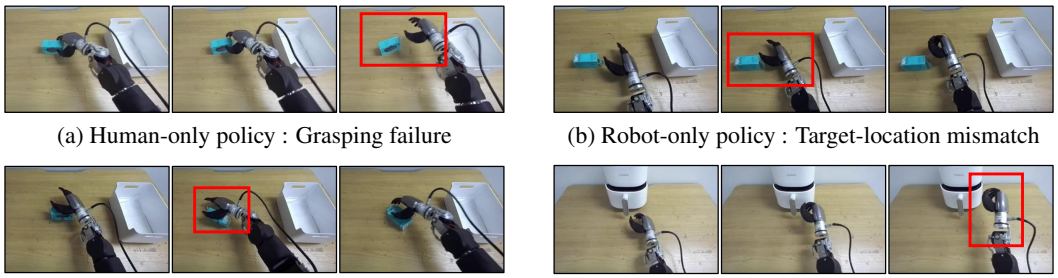
3.3 Scaling to heterogeneous human-robot data

In this experiment, we scale H2RB through a large-scale human-robot mid-training across multiple embodiments. We then adapt the mid-trained model to downstream robot tasks and compare it with the original VLA backbone directly fine-tuned on the same downstream task. Full mid-training details are provided in Appendix D.

As shown in Figure 4, mid-training improves object transfer from 51.8 to 56.9 (+5.1) and skill transfer from 62.7 to 84.7 (+22.0). These gains show that incorporating broader human experience during mid-training further improves downstream human-to-robot transfer, beyond task-specific co-training alone. Detailed task-wise results and training settings are provided in Appendix D.3.

3.4 Enhancing skill transfer with instruction conditioning

For skill-level transfer, we examine whether stronger language conditioning can improve the transferred behavior. We treat the instruction as an explicit skill signal by concatenating its representation with the state-conditioned vision-language latent and re-injecting it throughout the expert layers. As shown in Figure 5, this improves skill transfer from 62.7 to 83.3 (+20.6), suggesting that sustained



(a) Human-only policy : Grasping failure

(b) Robot-only policy : Target-location mismatch

(c) Ego-Pi-shared action : Invalid grasp pose

(d) Ego-Pi-shared action : Task-motion mismatch

Figure 6: **Failure case analysis of human-to-robot transfer.** Rollout examples compare the failure modes across human-only, robot-only, and Ego-Pi-shared action policies. (a) Fails to complete the grasp. (b) Unrelated to the target object location. (c) Fails to produce a hand pose suitable for grasping. (d) Executes a robot-observed pick-and-place motion in human-observed task environment.

instruction access helps preserve task semantics during cross-embodiment execution. Implementation details are provided in Appendix A.4.

4 Analysis

Human-to-robot transfer must combine human-provided task structure with robot-specific execution knowledge. Accordingly, H2RB reduces cross-embodiment mismatch in the conditioning and action-prediction spaces while preserving source-specific motion dynamics. We analyze this separation through transfer failures, execution speed, and the resulting human-robot representation.

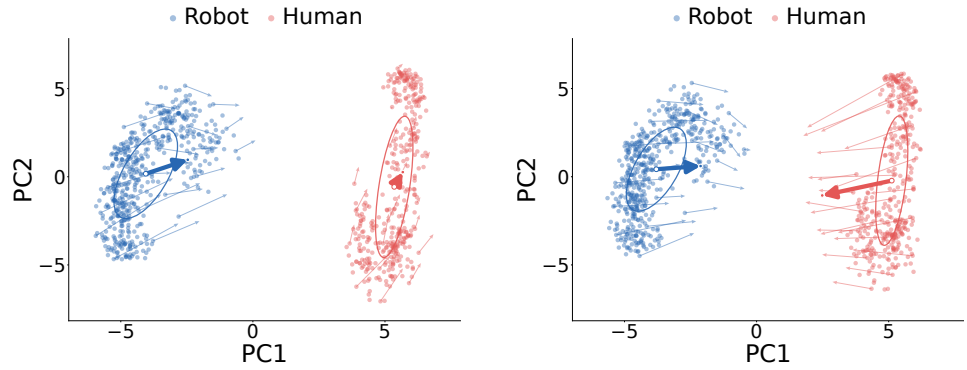
4.1 Qualitative analysis of transfer failures

Human demonstrations specify the behavior, but not fine-grained robot execution. As shown in Figure 6a, policies trained only on human demonstrations often capture the high-level interaction pattern, including the approach direction and action sequence, yet struggle with fine-grained actions near contact, such as precise wrist alignment, grasp pose, and hand configuration. This suggests that human demonstrations provide useful supervision for what interaction should be performed, but embodiment-dependent execution details can hinder successful transfer to the robot. As a result, the learned policy may reproduce the overall behavior while failing to acquire the precise contact-level control required by the target embodiment.

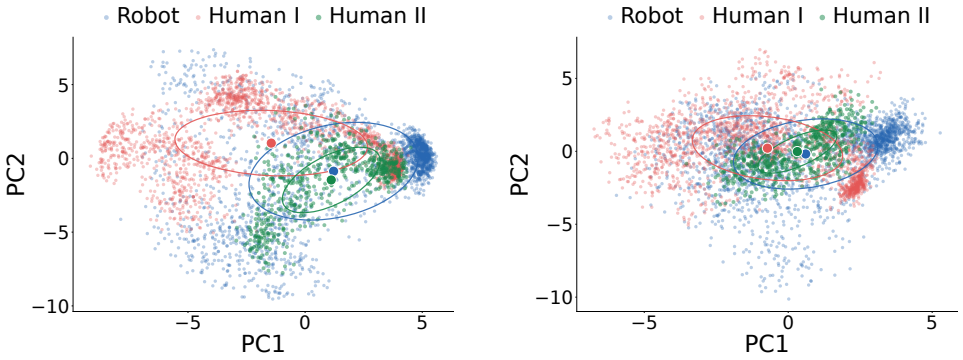
Robot demonstrations provide execution patterns, but not the target behavior. Robot-only policies produce robot-compatible wrist motion and hand closure, yet fail to reproduce behaviors introduced only through human demonstrations, such as approaching and interacting correctly with the human-demonstrated target object. As shown in Figure 6b, robot data provide embodiment-specific execution experience, but lack supervision for the unseen target behavior.

A shared action representation alone does not fully resolve the perceptual gap. Co-trained policies acquire grounding to human-demonstrated target objects, unlike the robot-only policy, but this grounding often lacks the spatial precision required for reliable execution. As shown in Figure 6c, the policy approaches the intended object yet produces inaccurate hand poses that fail to establish valid contact. These failures suggest that aligning human and robot data only in action space is insufficient; the policy must also learn perceptually compatible conditioning representations while preserving embodiment-specific execution.

A shared action decoder cannot resolve embodiment-specific execution differences. As shown in Figure 6d, the shared-decoder baseline fails to reproduce the human-demonstrated opening behavior and instead executes a robot-observed pick-and-place motion. At deployment, the policy receives only robot observations and states, producing latents closer to the robot domain; a shared decoder can therefore favor robot-associated action patterns despite being trained on both human and robot motions. H2RB instead uses embodiment-specific decoders shared across skills, allowing human-only behaviors to transfer through the common latent space while preserving embodiment-appropriate execution.



(a) Conditioning-space adaptation



(b) Action-latent adaptation

Figure 7: **PCA visualization of cross-embodiment representation adaptation.** (a) Vision-language conditioning features with hand geometry-anchored alignment alone (left) and H2RB (right). Arrows show adapter-induced displacements for the same observation. (b) Action-tokenizer latents without (left) and with (right) cross-domain latent regression, using per-tokenizer PCA. Human I and II denote human tasks trained in within-skill object transfer and skill transfer, respectively.

4.2 Cross-embodiment gap analysis

Hand geometry-anchored alignment contracts the conditioning space. Hand geometry-anchored alignment supervises human and robot observations through a common hand-geometry target, regardless of human-robot paired data. The PCA projection in Figure 7a qualitatively shows robot features shifting toward the human distribution, with decreased human-robot gap. Adding decoupled-decoder latent tokenizer further brings the two distributions closer. This suggests that hand geometry-anchored alignment already induces cross-embodiment compatibility in the conditioning space, which is further reinforced by learning a common action-prediction space.

Decoupled-decoder latent tokenizer progressively aligns the prediction space. Table 4a shows that introducing the action tokenizer improves object transfer from 34.7 to 43.3 and enables skill transfer from 0.0 to 49.7. Cross-domain regression further reduces the normalized human-robot latent gap from 3.81 to 1.53, a 59.8% reduction, while improving object and skill transfer to 51.8 and 62.7, respectively. At deployment, robot-conditioned predictions may remain closer to the robot latent distribution, even for tasks learned from human data. This creates a mismatch with the human latent distribution seen during human-decoder training. Cross-domain regression is therefore needed to reduce this shift by aligning the two latent distributions. The two-dimensional projections in Figure 7b provide a qualitative view of this alignment, showing how cross-domain regression brings the human and robot latent distributions closer in the action latent space.

Faster task execution from human data. Human demonstrations can provide faster task-specific motion while requiring substantially less collection time and robot interaction. To test whether these dynamics are preserved after transfer, we collected 100 robot teleoperation episodes for the same

Table 4: **Analysis of the decoupled-decoder latent tokenizer.** (a) Effect of the action tokenizer and cross-domain latent regression on object transfer, skill transfer, and normalized human-robot latent gap. (b) Motion speed and average success of embodiment-specific reconstruction decoders.

(a) Action tokenizer ablation

Action Tokenizer	Cross-domain latent regression	Object transfer	Skill transfer	Human-robot latent gap
✗	✗	34.7	0.0	-
✓	✗	43.3	49.7	3.81
✓	✓	**51.8**	**62.7**	**1.53**

(b) Embodiment-specific decoding

Training source	Motion speed (cm/s)	Weighted task score
Robot demonstrations	2.52	**65.3**
Human demonstrations	**7.36** (×2.92)	62.7

Open/Close tasks, with 50 episodes per task, and trained a robot-only policy. As shown in Table 4b, its rollouts reached an average wrist end-effector speed of 2.52 cm/s, whereas the policy transferred from human demonstrations reached 7.36 cm/s, a 2.92× increase, while retaining comparable task performance (62.7 vs. 65.3). Because the embodiment-specific decoders preserve distinct action dynamics, the transferred policy can retain the faster motion characteristics of human demonstrations when deployed on the robot.

5 Related Work

Learning from human demonstrations. Human videos provide policy supervision through reconstructed or tracked hand motion (Zhang et al., 2025a; Hoque et al., 2026; Li et al., 2025; Qin et al., 2022) or latent actions inferred from frame transitions (Ye et al., 2025; Chen et al., 2026), supporting co-training and pretraining in works such as EgoMimic, EgoVLA, and EgoScale (Kareer et al., 2025; Yang et al., 2025; Zheng et al., 2026; Tao et al., 2025). Ego-Pi (Kim et al., 2026b) transfers task semantics through co-training in a common wrist-hand space, while HumanEgo (Wang et al., 2026) learns from human demonstrations without robot data using hand-object representations. We focus on adapting a robot-pretrained VLA to human-demonstrated target tasks while retaining experience from robot data.

Cross-embodiment representation alignment. Human-data pretraining builds cross-embodiment priors (Yang et al., 2025; Zheng et al., 2026), while explicit alignment relies on paired demonstrations (Xie et al., 2026; Zhu et al., 2026) or trajectory-based pseudo-pairs (Punamiya et al., 2025). PHANTOM and Masquerade instead synthesize robot-domain observations through inpainting and rendering (Lepert et al., 2025b;a). H2RB grounds pretrained VLA features in hand geometry independently for each observation, providing a shared physical reference without paired data, trajectory matching, image synthesis, or prior human-robot backbone training.

Latent action representations and decoding. Diffusion and flow-based policies generate continuous actions (Chi et al., 2025; Black et al., 2024; Ze et al., 2024; Zhang et al., 2025b), while latent diffusion separates generation from output reconstruction (Rombach et al., 2022; Peebles & Xie, 2023). LAD aligns retargeted action pairs (Bauer et al., 2025), and XL-VLA learns cross-hand latent representations (Jiang et al., 2026), both retaining embodiment-specific decoders for heterogeneous end-effectors. H2RB extends this separation to human-robot demonstrations expressed in the same wrist-hand space, sharing latent prediction while preserving embodiment-specific motion dynamics through separate temporal decoders.

6 Conclusion

We present H2RB, a framework for adapting pretrained VLAs from unpaired human and robot demonstrations. Rather than relying on implicit cross-embodiment alignment, H2RB bridges perception through hand geometry-anchored alignment and action through a shared latent space with embodiment-specific decoders. Human demonstrations provide target behavior, while robot data supplies deployment-specific execution knowledge. On real-world transfer, H2RB improves in both object & skill transfer, with further gains from heterogeneous mid-training. These results highlight the value of sharing task-relevant structure without forcing human and robot motion dynamics to coincide. We hope our approach helps bridge scalable human demonstration data and robot task adaptation, enabling pretrained VLAs to acquire new behaviors with human demonstration data.

Reproducibility Statement

We provide implementation details, training hyperparameters, data and baseline definitions, evaluation protocols, and additional ablations in the Appendix. These include the action and geometry preprocessing, action tokenizer, policy optimization and deployment procedure, and evaluation accounting needed to reproduce our experiments.

Ethics Statement

Human demonstrations were collected by the authors and by employed data collectors for robotic manipulation research. The data are used solely as manipulation supervision and not to infer personal or sensitive attributes. All robot experiments are conducted in controlled research environments.

AI use statement

We used generative AI tools to assist with literature discovery, manuscript drafting and editing, and limited routine coding and visualization tasks. These tools were not used to develop the core research ideas, generate experimental results, produce synthetic datasets, or establish mathematical or empirical claims presented in this paper. All AI-assisted outputs were reviewed and verified by the authors, who take full responsibility for the final content of the manuscript.

References

Erik Bauer, Elvis Nava, and Robert K Katzschmann. Latent action diffusion for cross-embodiment manipulation. *arXiv preprint arXiv:2506.14608*, 2025.

Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. pi_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

Qingwen Bu, Yanting Yang, Jisong Cai, Shenyuan Gao, Guanghui Ren, Maoqing Yao, Ping Luo, and Hongyang Li. Learning to act anywhere with task-centric latent actions. In *Proceedings of Robotics: Science and Systems*, 2025. doi: 10.15607/RSS.2025.XXI.014.

Xiaoyu Chen, Hangxing Wei, Pushi Zhang, Chuheng Zhang, Kaixin Wang, Yanjiang Guo, Rushuai Yang, Yucen Wang, Xinquan Xiao, Li Zhao, et al. Villa-x: enhancing latent action modeling in vision-language-action models. In *International Conference on Learning Representations*, volume 2026, pp. 70673–70703, 2026.

Yi Chen, Yuying Ge, Weiliang Tang, Yizhuo Li, Yixiao Ge, Mingyu Ding, Ying Shan, and Xihui Liu. Moto: Latent motion token as the bridging language for learning robot manipulation from videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19752–19763, 2025.

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. *arXiv preprint arXiv:2402.10329*, 2024.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

Weisheng Dai, Kai Lan, Jianyi Zhou, Bo Zhao, Xiu Su, Junwen Tong, Weili Guan, and Shuo Yang. Conla: Contrastive latent action learning from human videos for robotic manipulation. *arXiv preprint arXiv:2602.00557*, 2026.

Hongming Fu, Wenjia Wang, Xiaozhen Qiao, Rolandos Alexandros Potamias, Taku Komura, Shuo Yang, Zheng Liu, and Bo Zhao. Egograsp: World-space hand-object interaction estimation from egocentric videos. *arXiv preprint arXiv:2601.01050*, 2026.

Ryan Hoque, Peide Huang, David Yoon, Jian Zhang, et al. Egodex: Learning dexterous manipulation from large-scale egocentric video. In *International Conference on Learning Representations*, volume 2026, pp. 4218–4237, 2026.

Guangqi Jiang, Yutong Liang, Jianglong Ye, Jia-Yang Huang, Changwei Jing, Rocky Duan, Pieter Abbeel, Xiaolong Wang, and Xueyan Zou. Cross-hand latent representation for vision-language-action models. *arXiv preprint arXiv:2603.10158*, 2026.

Simar Kareer, Dhruv Patel, Ryan Punamiya, Pranay Mathur, Shuo Cheng, Chen Wang, Judy Hoffman, and Danfei Xu. Egomimic: Scaling imitation learning via egocentric video. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 13226–13233. IEEE, 2025.

Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

Dongyoung Kim, Huiwon Jang, Myungkyu Koo, Suhyeok Jang, Taeyoung Kim, Beomjun Kim, Byungjun Yoon, Changsung Jang, Daewon Choi, Dongsu Han, et al. Rldx-1 technical report. *arXiv preprint arXiv:2605.03269*, 2026a.

Ji Woong Kim, Ke Wang, Zipeng Fu, Sirui Chen, Jeff Lai, Chelsea Finn, et al. Ego-pi: Vla finetuning for ego-centric human and robot data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1515–1524, 2026b.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

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

Marion Lepert, Jiaying Fang, and Jeannette Bohg. Phantom: Training robots without robots using only human videos. *arXiv preprint arXiv:2503.00779*, 2025b.

Qixiu Li, Yu Deng, Yaobo Liang, Lin Luo, Lei Zhou, Chengtang Yao, Lingqi Zeng, Zhiyuan Feng, Huizhi Liang, Sicheng Xu, et al. Scalable vision-language-action model pretraining for robotic manipulation with real-life human activity videos. *arXiv preprint arXiv:2510.21571*, 2025.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4172–4182. IEEE, 2023.

Ryan Punamiya, Dhruv Patel, Patcharapong Aphiwetsa, Pranav Kuppili, Lawrence Y Zhu, Simar Kareer, Judy Hoffman, and Danfei Xu. Egobridge: Domain adaptation for generalizable imitation from egocentric human data. In *Human to Robot: Workshop on Sensorizing, Modeling, and Learning from Humans*, 2025.

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*, pp. 570–587. Springer, 2022. doi: 10.1007/978-3-031-19842-7_33.

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.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 10674–10685. ieee, 2022.

Tony Tao, Mohan Kumar Srirama, Jason Jingzhou Liu, Kenneth Shaw, and Deepak Pathak. Dexwild: Dexterous human interactions for in-the-wild robot policies. In *Proceedings of Robotics: Science and Systems*, 2025. doi: 10.15607/RSS.2025.XXI.075.

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 *Proceedings of the 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 201–221. PMLR, 2023.

Zhi Wang, Botao He, Kelin Yu, Seungjae Lee, Ruohan Gao, Furong Huang, and Yiannis Aloimonos. Humanego: Zero-shot robot learning from minutes of human egocentric videos. *arXiv preprint arXiv:2605.24934*, 2026.

Sicheng Xie, Haidong Cao, Zejia Weng, Zhen Xing, Haoran Chen, Shiwei Shen, Jiaqi Leng, Zuxuan Wu, and Yu-Gang Jiang. Human2robot: Learning robot actions from paired human-robot videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 11078–11086, 2026.

Ruihan Yang, Qinxi Yu, Yecheng Wu, Rui Yan, Borui Li, An-Chieh Cheng, Xueyan Zou, Yunhao Fang, Xuxin Cheng, Ri-Zhao Qiu, et al. Egovla: Learning vision-language-action models from egocentric human videos. *arXiv preprint arXiv:2507.12440*, 2025.

Seonghyeon Ye, Joel Jang, Byeongguk Jeon, Se June Joo, Jianwei Yang, Baolin Peng, Ajay Mandlekar, Reuben Tan, Yu-Wei Chao, Bill Yuchen Lin, et al. Latent action pretraining from videos. In *International Conference on Learning Representations*, volume 2025, pp. 28213–28239, 2025.

Yanjie Ze, Gu Zhang, Kangning Zhang, Chenyuan Hu, Muhan Wang, and Huazhe Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. In *Proceedings of Robotics: Science and Systems*, 2024. doi: 10.15607/RSS.2024.XX.067.

Jinglei Zhang, Jiankang Deng, Chao Ma, and Rolandos Alexandros Potamias. Hawor: World-space hand motion reconstruction from egocentric videos. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1805–1815. IEEE, 2025a.

Qinglun Zhang, Zhen Liu, Haoqiang Fan, Guanghui Liu, Bing Zeng, and Shuaicheng Liu. Flowpolicy: Enabling fast and robust 3d flow-based policy via consistency flow matching for robot manipulation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 14754–14762, 2025b. doi: 10.1609/aaai.v39i14.33617.

Ruijie Zheng, Dantong Niu, Yuqi Xie, Jing Wang, Mengda Xu, Yunfan Jiang, Fernando Castañeda, Fengyuan Hu, You Liang Tan, Letian Fu, et al. Egoscale: Scaling dexterous manipulation with diverse egocentric human data. *arXiv preprint arXiv:2602.16710*, 2026.

Xiang Zhu, Puzhen Yuan, Yichen Liu, and Jianyu Chen. Harp-vla: Human-robot aligned representation learning for vision-language-action model. *arXiv preprint arXiv:2605.31234*, 2026.

A Implementation details

A.1 Action and geometry preprocessing

Physical actions and current geometry. A single-hand OpenArm action concatenates a three-dimensional wrist position, a six-dimensional rotation representation, and six Inspire-hand coordinates, giving 15 values per timestep. Current hand geometry concatenates five wrist-local fingertip positions with the wrist pose, giving 24 values. These are different supervision targets: action learning uses future commanded wrist poses and hand coordinates, whereas hand geometry-anchored alignment uses the currently observed hand configuration. The bimanual wrist-hand representation concatenates the two hands to give 30-dimensional actions and 48-dimensional geometry.

Human demonstrations. Human videos are processed with a HaWoR-based reconstruction pipeline (Zhang et al., 2025a). Reconstructed wrist motion is converted to the robot coordinate convention, and hand articulation is retargeted to the shared wrist-hand action representation (Kim et al., 2026b). The reconstructed observed hand also supplies current geometry targets. Robot demonstrations provide wrist targets and hand commands directly, while robot geometry targets are obtained from the observed configuration.

Table 5: Single-hand architecture and representations.

Component	Configuration
Backbone	RLDX-1-IMG (Kim et al., 2026a)
Policy inputs	One current egocentric RGB frame, state, and instruction
Conditioning representation	64 cognition tokens
Physical action / current geometry	15 / 24 coordinates per hand
Data rate / action chunk	20 Hz / $K = 16$ timesteps
Latent action target	16×64 posterior-mean coordinates
Geometry adapter	Token-wise residual MLP; hidden width 256; SiLU
Adapter output initialization	Zero projection weights and bias
Geometry readout	Mean pooling followed by a linear map
Tokenizer encoder / each decoder	4 temporal blocks; width 256; 4 attention heads
Default transfer decoder	Frozen human reconstruction decoder

Normalization. Human retargeted actions are first converted to the canonical robot frame. Tokenizer inputs are normalized coordinate-wise as $\tilde{a}_j = \mathrm{clip}(2(a_j - m_j)/(M_j - m_j) - 1, -1, 1)$, where m_j and M_j are the minimum and maximum across the human and robot source statistics. The same rule applies separately to wrist position, the six rotation components, and hand joint angles. Both reconstruction decoders share these bounds and use $\hat{a}_j = m_j + \frac{1}{2}(\mathrm{clip}(\hat{\tilde{a}}_j, -1, 1) + 1)(M_j - m_j)$. Their outputs are already in the canonical Ego-pi-shared action frame, so the human-to-robot conversion is not applied again.

Policy states are normalized independently using the same affine formula, with lower and upper bounds given by the minimum source 1st percentile and maximum source 99th percentile, respectively. These are shared bounds within the common embodiment tag, not percentiles recomputed from pooled samples.

A.2 Decoupled-decoder latent tokenizer architecture and objectives

Shared encoder. The latent tokenizer projects each 15-dimensional action to width 256, adds fixed sinusoidal temporal embeddings, and applies four bidirectional temporal Transformer blocks. Each block uses four attention heads of width 64, pre-layer normalization, query/key normalization, and a SiLU-gated feed-forward network. A final layer normalization and a $256 \to 64$ projection produce one posterior-mean token per timestep. A $64 \to 64$ head predicts log-variance from the mean, clipped to $[-8, 8]$. No image, visual-feature target, or domain token enters the encoder.

Embodiment-specific reconstruction decoders. Each complete decoder uses a $64 \to 256$ up-projection, sinusoidal positions, four Transformer blocks of the same type as the encoder, and a non-affine final layer normalization followed by a $256 \to 15$ output head. The human decoder is initialized from the robot decoder before embodiment-specific reconstruction training. The branches have independent temporal blocks and output heads, not merely separate final linear layers. Both tokenizers are trained from scratch for 100k steps: Arm I uses robot bottle/cup/doll/snack and human box/ball/towel pick-and-place demonstrations; Arm II uses robot box/ball/towel pick-and-place and human air-fryer opening/closing demonstrations. Decoder copying occurs at random initialization, with no pretrained tokenizer checkpoint or additional warm-start stage. Both use SiLU-gated feed-forward layers with nominal expansion ratio; all dropout and stochastic depth are zero.

Reconstruction and KL reductions. For valid action coordinates, the tokenizer minimizes absolute reconstruction error through the decoder selected by the example's source. The Gaussian KL penalty sums $\frac{1}{2}(\mu^2 + \exp(\ell) - 1 - \ell)$ over latent coordinates, where ℓ denotes log-variance. Reconstruction and cross-domain latent regularization use the same reparameterized sample during tokenizer training. The objective is

$$\mathcal{L}_{\text{tok}} = \mathcal{L}_{\text{rec}} + 10^{-6}\mathcal{L}_{\text{KL}} + 0.03\big(\mathcal{L}_{\text{cent}} + \mathcal{L}_{\text{var}} + 0.004\mathcal{L}_{\text{cov}}\big). \tag{4}$$

Reconstruction is averaged over examples, all 16 temporal positions, and all 15 action coordinates; no coordinate padding is used, and end-of-episode repetitions remain included. KL is averaged over examples and temporal positions but summed over the 64 latent dimensions. Human and robot examples receive equal per-example weight, without additional source balancing; their frequencies follow dataset-size-proportional sampling.

Position-wise centroid regression. For a minibatch containing complete action chunks from both sources, let $m_{d,k}$ be the mean sampled latent from source $d \in \{H, R\}$ at relative chunk position k. With $K = 16$ and $d_z = 64$, centroid regression is

$$\mathcal{L}_{\text{cent}} = \frac{1}{K d_z} \sum_{k=1}^{K} \|m_{H,k} - m_{R,k}\|_2^2. \tag{5}$$

Relative chunk positions are not matched semantic stages. A valid bin requires at least two human and two robot examples. Valid bins are weighted by the smaller source count. With complete length-16 chunks, the reduction is Eq. (5).

Variance and covariance regularization. Let $X_d \in \mathbb{R}^{n_d \times 64}$ stack latent tokens from source d after cross-device gathering. The variance term uses a population variance reduction:

$$\mathcal{L}_{\text{var}} = \frac{1}{2 \cdot 64} \sum_{d \in \{H,R\}} \sum_{j=1}^{64} \max\left(0, 1.75 - \sqrt{\text{Var}(X_{d,:,j}) + 10^{-4}}\right)^2. \tag{6}$$

For centered sample covariance $\Sigma_d = (X_d - \overline{X}_d)^\top (X_d - \overline{X}_d)/(n_d - 1)$,

$$\mathcal{L}_{\text{cov}} = \frac{1}{2 \cdot 64} \sum_{d \in \{H,R\}} \sum_{j \neq j'} (\Sigma_d)_{j,j'}^2. \tag{7}$$

Variance and covariance are computed globally within each source rather than independently per semantic phase. These safeguards oppose collapse but do not guarantee correct sample-level human-robot correspondence.

Frozen policy targets. After tokenizer training, the encoder and both decoders are frozen. Policy targets are cached posterior means, not fresh VAE samples, and use no additional latent standardization.

A.3 Policy optimization and deployment

Hand geometry-anchored alignment. For conditioning tokens $\mathbf{h}_t$, the adapted sequence is $\mathbf{h}_t^{\text{geo}} = \mathbf{h}_t + A_\phi(\mathbf{h}_t)$. The linear readout acts on its mean-pooled representation, while the action expert receives the full adapted token sequence. For sample i, coordinate-validity mask $m_{i,j}$,

and unit-transition SmoothL1 penalty ρ, the per-example loss is

$$\ell_{\text{geo}}(i) = \frac{\sum_j m_{i,j}\rho(\widehat{g}_{i,j} - g_{i,j})}{\max(1, \sum_j m_{i,j})}. \tag{8}$$

The batch loss averages examples with at least one valid coordinate. The geometric readout is used only during training, with weight $\lambda_{\text{geo}} = 0.2$ in the joint VLA objective.

Latent flow matching. We implement $\mathcal{L}_{\text{pred}}$ using conditional flow matching (Lipman et al., 2022). For a cached posterior-mean target $\mathbf{Z}_t^\star$, Gaussian noise $\boldsymbol{\varepsilon}$, and flow time τ sampled with the backbone's training-time distribution,

$$\mathbf{Z}_\tau = (1-\tau)\boldsymbol{\varepsilon} + \tau\mathbf{Z}_t^\star, \qquad \mathbf{V}^\star = \mathbf{Z}_t^\star - \boldsymbol{\varepsilon}. \tag{9}$$

Using $\mathbf{C}_t$ for the conditioning sequence supplied to the expert,

$$\mathcal{L}_{\text{pred}} = \mathbb{E}\left[\frac{1}{K d_z}\|v_\theta(\mathbf{Z}_\tau, \tau, \mathbf{s}_t, \mathbf{C}_t) - \mathbf{V}^\star\|_F^2\right], \qquad d_z = 64. \tag{10}$$

Here v_θ is a velocity predictor. The latent policy π_θ in Eq. (3) is implemented by numerical integration of this field from Gaussian noise. The shared wrist-hand baseline and its geometry-grounded variant apply the prediction objective to valid physical action coordinates instead of tokenizer latents. We sample $u \sim \text{Beta}(1.5, 1)$ and set $\tau = 0.999(1-u)$, with no further time rescaling. The backbone (RLDX-1-IMG) action expert uses four double-stream and eight single-stream blocks with 24 attention heads of dimension 64. Without the additional instruction pathway, $\mathbf{C}_t$ consists of 64 layer-normalized VLM latent tokens conditioned on the image and instruction; the hand geometry-anchored baseline and H2RB additionally apply a residual adapter with a 256-dimensional SiLU bottleneck.

Table 6: Optimization settings for tokenizer training, task-specific fine-tuning (FT), and heterogeneous mid-training (MT).

Setting	Tokenizer	Single-hand FT	Heterogeneous MT
Updates	100k	30k	100k
Global batch size	512	64	128
Optimizer	AdamW	AdamW	AdamW
Learning rate	(5×10^{-5})	10^{-4}	10^{-4}
Schedule / warmup	constant / None	Cosine / 5%	constant / None
Weight decay	10^{-5}	10^{-5}	10^{-5}
Gradient clipping	1.0	1.0	1.0
Precision	BF16	BF16	BF16
Seed	42	42	42
Geometry-loss weight	Not applicable	0.2	0.2

Table 7: Parameter updates during policy learning. Tokenizer training is a separate stage; its encoder and decoders are frozen during single-hand FT.

Module	Single-hand FT	Heterogeneous MT
Vision encoder	Frozen	Frozen
Language model	Frozen	Top four layers trainable
Cognition-token parameters	Frozen	Trainable
Geometry adapter / readout	Trainable / Trainable	Trainable / Trainable
Action projector / expert	Trainable	Trainable
State encoder	Trainable	Trainable
Tokenizer encoder / decoders	Frozen	Frozen where used

Fine-tuning and data sampling. All single-hand policies start from the same RLDX-1-IMG pre-trained checkpoint and are fine-tuned for 30k updates with global batch size 64. Human-robot

co-training uses equal domain weights and uniform task weights within each domain; single-source baselines use uniform task weights. Sampling cycles deterministically through action windows, without enforcing an exact domain split in each minibatch.

Within-skill and object transfer use separately trained tokenizers, each trained for 100k updates outside the policy-FT budget. Tokenizer sampling is proportional to training-window counts. The tokenizer recipe uses a fixed episode-level validation split; policy FT uses no separate validation set.

Robot deployment. The decoder is fixed across target objects rather than switched per object. We generate 16-action chunks using four explicit Euler steps. Decoded actions are clipped to the tokenizer's normalized range before inverse normalization. Wrist poses are converted to arm commands using Mink IK with the DAQP solver and model-defined joint limits, initialized from the observed joint state; hand joint commands are applied directly.

A.4 Additional instruction conditioning

We evaluate instruction renewal as H2RB + Inst., separately from the base H2RB. Instruction features are extracted from the frozen VLM at text positions preceding the image, so they encode the instruction without attending to visual tokens. After trainable layer normalization, these features are concatenated with the adapted vision-language cognition tokens at the action-expert input.

Before every action-expert block, the instruction slots are restored to their initial features, using the existing stream projection where required, while the other token representations evolve normally. We compare the full instruction sequence with its last valid token; the input-only control uses the full sequence without restoration. Padding positions are masked throughout.

B Real-world data and evaluation

Hardware and observations. Our real-world experiments use an OpenArm bimanual platform equipped with two Inspire RH56F1 dexterous hands. The controlled transfer experiments use only the right arm and hand. A head-mounted ZED 2i camera provides 1280×720 RGB observations from a viewpoint approximately matching the human demonstrations. Policies receive one current right-camera RGB frame, a language instruction, and a 15-dimensional proprioceptive state comprising wrist position (3D), wrist rotation (6D), and hand joint angles (6D). Wrist poses are computed by forward kinematics in a fixed canonical camera frame. Rollouts run at 20Hz.

B.1 Training data and baseline definitions

Transfer settings. Within-skill object transfer introduces Box, Ball, and Towel pick-and-place through human demonstrations while retaining source robot pick-and-place experience. Skill-level transfer introduces air-fryer opening and closing, which are absent from the source robot demonstrations. The corresponding robot in-domain evaluation tasks are Doll PnP and Box PnP, respectively. Human and robot data are collected in the same environment, without frame-level human-robot correspondence.

Table 8: Task and training-data conditions. PnP denotes pick-and-place. Policy FT uses all retained demonstrations without a held-out validation split; evaluation uses robot rollouts.

Item	Object transfer	Skill transfer
Human target tasks	Box / Ball / Towel PnP	Air-fryer Open / Close
Robot in-domain evaluation	Doll PnP	Box PnP
Exact robot training task list	Bottle / Cup / Doll / Snack PnP	Box / Ball / Towel PnP
Retained robot episodes	208 (64 / 48 / 32 / 64)	300 (100 / 100 / 100)
Retained human episodes per task	Box: 100; Ball: 100; Towel: 100	Open: 50; Close: 50

Available corpus accounting. The recorded single-hand corpus snapshot contains 908 episodes and approximately 208,600 frames at 20 Hz. It comprises 508 robot pick-and-place episodes, 300

human object-transfer episodes, and 100 human opening/closing episodes. The robot Open/Close reference is a separate collection of 50 demonstrations per task.

Baseline scope. "Human-only" and "in-domain-only" describe adaptation data, not the absence of prior knowledge in the common robot-pretrained backbone. The target-robot-supervised Open/Close reference used for the speed comparison is distinct from the source-task-only robot baseline in Table 1 and Table 2.

Table 9: Baseline and variant definitions using the interface names in the main comparison.

Condition	Adaptation sources	Action / conditioning interface
Base policy human-only	Human targets	Human demonstrations only, using HaWoR-extracted fingertip and wrist motions as the action representation.
Base policy in-domain-only	Source robot	Robot demonstrations only, using the native LeRobot robot action representation.
Base policy co-train	Source robot + human targets	Human and robot data retain their respective action interfaces and are trained with separate embodiment IDs. At inference, the target robot embodiment ID is used.
HaWoR-shared action (Zhang et al., 2025a)	Source robot + human targets	Shared fingertip and wrist action representation. Human actions are obtained directly from HaWoR, while robot actions are converted to the same representation through forward kinematics; both use the same action decoder.
Ego-Pi-shared action (Kim et al., 2026b)	Source robot + human targets	Shared physical wrist-hand actions; no proposed geometry grounding or latent tokenizer.
Shared action + hand geometry-anchored alignment	Source robot + human targets	Shared physical action prediction with hand geometry-anchored alignment.
H2RB	Source robot + human targets	Hand geometry-anchored alignment and a decoupled-decoder latent tokenizer with embodiment-specific decoders.
H2RB + Inst.	Source robot + human targets	H2RB with the additional instruction-renewal pathway.

B.2 Task-progress scoring

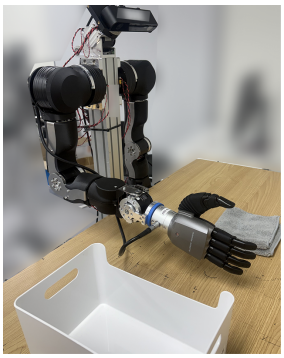
We report a weighted task score to capture partial execution as well as full completion. For n rollouts and the highest progress level w_i reached by rollout i,

$$S_{\mathrm{w}} = \frac{100}{n} \sum_{i=1}^{n} w_i. \tag{11}$$

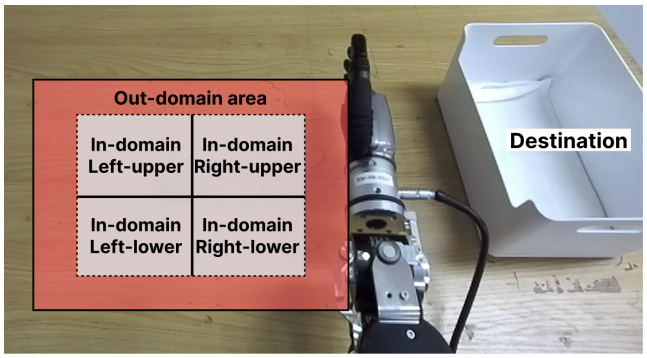
Pick-and-place uses levels 0 for failure, 0.3 for reach, 0.6 for grasp, 0.8 for moving the object, and 1.0 for full completion. Opening and closing use progress levels on the same scale, with task-specific criteria.

Table 10: Task-progress weights and milestones.

w_i	Pick-and-place	Open	Close
0.0	Failure	Failure	Failure
0.3	Reach to object	Reach to handle	Reach to handle
0.6	Valid grasping of object	Total grasp of handle	Total grasp of handle
0.8	The object lifted up	Opened halfway	Closed halfway
1.0	Complete task	Complete task	Complete task



(a) Robot setup



(b) Within-skill object transfer placement regions

Figure 8: **Robot setup and evaluation distributions.** **(a)** OpenArm robot platform used for real-world evaluation. **(b)** For object transfer, in-distribution (ID) rollouts are sampled from four workspace quadrants (right-upper, right-lower, left-upper, and left-lower), with five rollouts per quadrant and randomized object orientation. Out-of-distribution (OOD) rollouts are sampled outside the ID placement region (red area), again with randomized orientation. For skill transfer, the initial openness of the air fryer is randomized for all 30 trials.

B.3 ID/OOD PLACEMENT AND ROLLOUT PROTOCOL

For each target object, the object-transfer evaluation contains 20 in-distribution (ID) and 10 out-of-distribution (OOD) placement trials, totaling 90 target-task rollouts per policy. Skill transfer contains 15 Open and 15 Close trials, totaling 30. Robot in-domain performance is evaluated separately with 20 trials on the corresponding source task.

Table 11: Evaluation counts per policy in the main transfer comparison. The robot in-domain trials are separate from the target-task averages.

Evaluation group	ID	OOD	Total
I: Object transfer			
Object target: Box PnP	20	10	30
Object target: Ball PnP	20	10	30
Object target: Towel PnP	20	10	30
Object target subtotal	**60**	**30**	**90**
Robot in-domain: Doll PnP	Separate source-task evaluation		20
II: Skill transfer			
Skill target: Open	No ID/OOD split		15
Skill target: Close	No ID/OOD split		15
Skill target subtotal	**Open + Close**		**30**
Robot in-domain: Box PnP	Separate source-task evaluation		20

Placement distribution and evaluation protocol. For object transfer, we divide the nominal workspace into four quadrants-right-upper (RU), right-lower (RD), left-upper (LU), and left-lower (LD)-and sample five initial object placements from each quadrant, yielding 20 in-distribution (ID) rollouts per object. Object orientation is randomized independently for each rollout. We use the same quadrant-based placement protocol during human data collection, such that the ID evaluation distribution follows the spatial support observed during training. For out-of-distribution (OOD) evaluation, we instead sample 10 initial placements outside these quadrants, while continuing to randomize object orientation. For skill transfer, the initial state of the air fryer is randomized by varying its degree of opening for both the Open and Close tasks. Figure 8 illustrates the corresponding ID and OOD placement regions.

B.4 Evaluation aggregation and accounting

Because Box, Ball, and Towel use equal numbers of trials, the reported target-task average is the mean of the three per-object scores and is equivalently the weighted task score pooled over all 90 target-task rollouts. The corresponding pooled ID/OOD score is

$$S_{\mathrm{w}}^{\mathrm{all}} = \frac{60S_{\mathrm{w}}^{\mathrm{ID}} + 30S_{\mathrm{w}}^{\mathrm{OOD}}}{90}.$$

For skill transfer, Open and Close are each evaluated with 15 rollouts, so their reported average is equivalently the score pooled over all 30 target-task rollouts. The separate 20-rollout robot in-domain evaluation is not included in either target-task average.

All aggregate scores are computed from the per-rollout progress annotations before rounding; table entries are rounded to one decimal place.

We evaluate the final 30k-step policy checkpoint without validation-based checkpoint selection. Each reported condition uses one independently trained policy. All scheduled evaluation rollouts are included in the reported scores; failed task executions remain failures and are not retried or excluded.

C Additional results and ablations

C.1 Cumulative task progress

Table 12 reports cumulative target-task progress for the shared wrist-hand baseline and H2RB. Thresholds refer to the rubric in Section B.2. The weighted scores are the corresponding main-table values.

Table 12: Cumulative target-task progress (%) and weighted task scores (0-100). $R_{\geq 0.3}$, $R_{\geq 0.6}$, $R_{\geq 0.8}$, and $R_{=1.0}$ denote increasing completion thresholds; the last is full-task success.

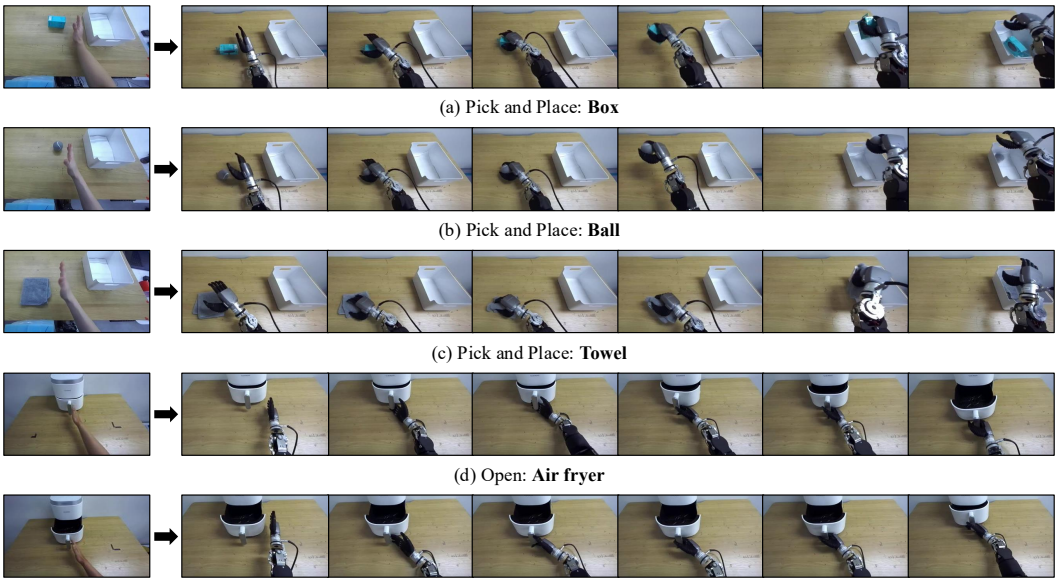
Policy	n	$R_{\geq 0.3}$	$R_{\geq 0.6}$	$R_{\geq 0.8}$	$R_{=1.0}$	S_{w}
I: Object transfer						
Ego-Pi-shared action	90	68.9	12.2	12.2	2.2	27.2
H2RB	90	95.6	46.7	33.3	12.2	51.8
II: Skill transfer						
Ego-Pi-shared action	30	0.0	0.0	0.0	0.0	0.0
H2RB	30	100.0	66.7	33.3	30.0	62.7

C.2 Hand geometry-anchored alignment under placement shift

Table 13 retains the intermediate hand geometry-anchored alignment with physical-action condition, in addition to the shared-action baseline and full method. ID and OOD counts follow Section B.3.

Table 13: Effect of geometry grounding on object transfer. Robot in-domain evaluates the familiar Doll PnP task, while human transfer reports weighted task scores under in-distribution (ID) and out-of-distribution (OOD) object placements. All values are weighted task scores (0-100).

	Robot in-domain	Human transfer		
Method	Doll PnP	ID	OOD	Avg.
Base policy co-train + Ego-Pi-shared action (Kim et al., 2026b)	30.0	30.2	21.3	27.2
+ Hand geometry-anchored alignment	47.5	36.0	32.0	34.7
+ H2RB (Ours)	**49.0**	**55.0**	**45.3**	**51.8**



(a) Pick and Place: **Box**

(b) Pick and Place: **Ball**

(c) Pick and Place: **Towel**

(d) Open: **Air fryer**

(e) Close: **Air fryer**

Figure 9: **Qualitative transferred-task rollouts.** Representative successful H2RB executions on box, ball, towel, open, and close, spanning both object-level and skill-level transfer.

C.3 Decoder selection ablations

Inference decoder selection. Both decoder branches are evaluated using the same within-skill object-transfer protocol described in Section B.3, with 30 rollouts per object and the same ID/OOD evaluation split. Only the reconstruction branch used at inference is changed; the latent policy and all other evaluation settings remain fixed. As shown in Table 14, the human decoder trained on human action data achieves better transfer performance than the robot decoder. The human decoder is therefore used for all main transfer results, without per-object decoder switching.

Table 14: Inference-decoder selection for the within-skill object transfer latent policy. Values are weighted task scores (0-100)

Reconstruction decoder	Box	Ball	Towel	Avg.
Robot decoder	37.0	16.0	45.0	32.7
Human decoder	**53.0**	**49.0**	**53.3**	**51.8**

C.4 Execution dynamics

To compare source-specific motion dynamics, we additionally train a robot-supervised reference on 50 Open and 50 Close demonstrations. As reported in Table 4b, the human-transfer policy reaches an average wrist speed of 7.36 cm/s, compared with 2.52 cm/s for the robot-supervised reference. The two policies are trained separately; this comparison is not an inference-decoder swap. Wrist speed is computed from consecutive policy-predicted wrist positions at 20 Hz over each full successful rollout, including stationary periods, then averaged across episodes within each task and equally across Open and Close.

D Heterogeneous multi-embodiment mid-training

The VLA backbone and hand geometry-anchored alignment are shared, while group-specific state and action interfaces accommodate different embodiments and action representations.

D.1 Data mixture and action interfaces

The mixture contains 14,548 episodes and 5,014,530 frames. In-house ALLEX robot data contain 480 demonstrations. In-house human data combine 1,480 head-mounted camera demonstrations with 930 demonstrations recorded by OpenArm ZED 2i camera. Single-hand OpenArm combines robot and human pick-and-place data with human air-fryer opening/closing demonstrations; EgoDex and bimanual OpenArm provide the remaining data. The ALLEX collections use 30 Hz, and all other collections use 20 Hz. Overall source combination is described in Table 15.

Table 15: Mid-training data by collection source.

Data source	Episodes	Frames
EgoDex (Hoque et al., 2026)	10,000	1,714,342
In-house human data	2,410	1,754,308
Single-hand OpenArm	1,035	236,751
Bimanual OpenArm	623	249,488
In-house ALLEX robot data	480	1,059,641
Total	14,548	5,014,530

We initialize five new state/action MLP interfaces while retaining the 36 pretrained interfaces of RLDX-1-IMG. In-house ALLEX robot and in-house human data share the 50-D action interface and its frozen tokenizer; single-hand OpenArm uses a separate 15-D tokenizer. EgoDex and bimanual OpenArm predict native 30-D wrist-hand actions, normalized using group-specific first and 99th percentiles with invalid coordinates masked. Tokenizer-based interfaces retain their own action normalization. All interfaces use 16-step action chunks and current egocentric images; ALLEX uses stereo views, whereas the other collections use a single view.

D.2 Optimization and downstream adaptation

State dropout is 0.3, and training images receive color jitter with brightness, contrast, and saturation strength 0.6 and hue strength 0.16.

Shared hand geometry-anchored alignment. A residual adapter of hidden width 256 and a shared linear readout predict 48-dimensional wrist-fingertip geometry, with 24 coordinates per hand. ALLEX targets are obtained through forward kinematics; the single-hand OpenArm group supervises only the right-hand coordinates and masks the left hand. The remaining groups use observed bimanual geometry. Geometry grounding has weight 0.2 and uses the masked loss in Eq. (8), regardless of whether the action target is latent or native. The backbone's auxiliary physics-loss coefficient is 0.1.

For downstream evaluation, both the original and mid-trained backbones are fine-tuned on the same OpenArm object- and skill-transfer tasks using the protocol in Section B.

D.3 Task-level transfer results

Table 16 and Table 17 report weighted task scores. This comparison measures the benefit of additional heterogeneous MT, rather than isolating hand geometry-anchored alignment within a matched MT recipe.

Table 16: Effect of heterogeneous mid-training on object transfer. Robot in-domain reports performance on a robot-data-trained task, while Human transfer evaluates target tasks introduced through human demonstrations. Each entry is the weighted task score (0-100).

	Robot in-domain	Human transfer			
Method	Doll	Box	Ball	Towel	Avg.
H2RB	49.0	53.0	49.0	53.3	51.8
MT + H2RB	**69.5**	**55.7**	**52.3**	**62.7**	**56.9**

Table 17: Effect of heterogeneous mid-training on skill transfer. Robot in-domain reports performance on a robot-data-trained task, while Human transfer evaluates target tasks introduced through human demonstrations. Each entry is the weighted task score (0-100).

	Robot in-domain	Human transfer		
Method	Box PnP	Open	Close	Avg.
H2RB	**81.0**	50.7	74.7	62.7
MT + H2RB	<u>80.5</u>	**72.0**	**97.3**	**84.7**