EGOWAM: World Action Models Beyond Pixels with In-the-Wild Egocentric Human Data

Baoyu Li*, **Xinchen Yin***, **Mengying Lin, Yixin Zhang, Danfei Xu**
Georgia Institute of Technology

Abstract: Egocentric human data offers scalable supervision for robot manipulation. However, behavior cloning entangles transferable content like objects, scenes, and task semantics, with non-transferable factors like human morphology, head motion, and behavioral style. We study whether World Action Models (WAMs) provide a better training signal by requiring policies to predict not only actions, but also how the scene evolves. The central question is what *world representation* best enables human-to-robot transfer. We hypothesize that an effective world target should abstract appearance, capture agent-invariant physical effects, and separate camera motion from environment change. We introduce EGOWAM, a controlled human-robot co-training framework that fixes the policy backbone, action head, and data mixture while varying only the world prediction target, comparing Pixel, DINO, and 3D motion flow. Across three real-world bimanual tasks, WAM co-training scales more effectively with in-the-wild egocentric human data than behavior cloning. Pixel-based prediction transfers weakly, while DINO and 3D flow yield substantial gains: DINO improves out-of-distribution object and scene generalization by up to 4×, and 3D flow improves in-domain performance by 20–30%. More details: gatech-rl2.github.io/egowam.github.io.

Keywords: World Action Models, Learn from Human Data, Robot Manipulation

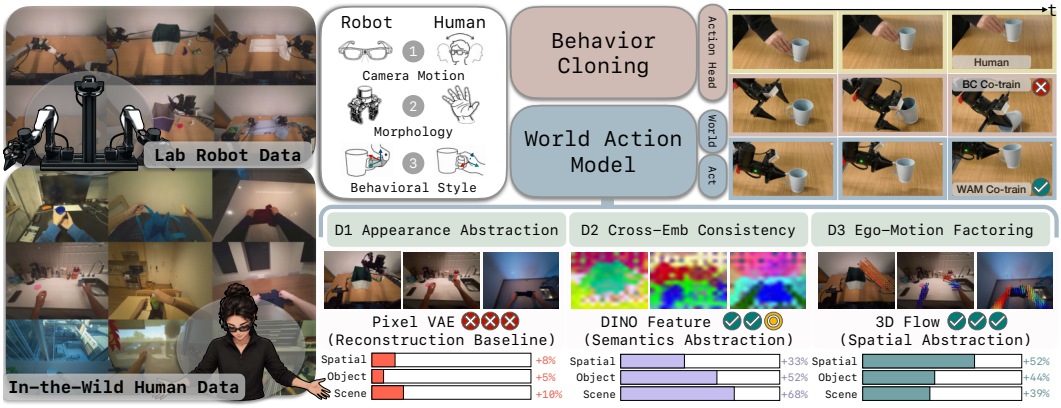


Figure 1: **EGOWAM** co-trains on lab robot data and *in-the-wild egocentric human data*, separated by an embodiment gap of camera motion, morphology, and behavioral style. **(1)** BC co-training leaks this gap through the *sole* action head as inexecutable motions that harm performance; WAM co-training adds a channel predicting scene evolution, transferring human data through dynamics where actions cannot. **(2)** EGOWAM studies *what world representation* best enables transfer, comparing pixel VAE, DINO features, and 3D flow against three desiderata: appearance abstraction, cross-embodiment consistency, and ego-motion factoring.

1 Introduction

Egocentric human data offers a promising source for scaling robot manipulation, providing diversity in objects, scenes, and behaviors that is prohibitively expensive to collect on robots [1, 2, 3]. A

* denotes equal contribution. Correspondence: bli678@gatech.edu.

growing body of work exploits this through behavior-cloning (BC) co-training, retargeting human demonstrations into a robot-compatible action space and jointly training an imitation policy [4, 5]. Recent efforts [6, 7] show this paradigm can scale, but only with the human data that is carefully aligned to the robot in viewpoint, motion speed, and behavioral style. Without that bridge, action-level co-training injects human-like motions that the robot cannot execute and *degrades* downstream performance (Fig. 1). This is the *bitter lesson* of action-level co-training: the shared action decoder is the *sole* channel through which human data reaches the policy, forcing it to entangle transferable content (objects, scenes, semantics) with non-transferable execution (morphology, behavioral style), and the embodiment gap in the latter blocks transfer of the former.

World Action Models (WAMs) [8, 9, 10, 11] open a second supervision channel: an auxiliary world-model head predicts future states from a shared backbone, grounding action prediction in task-relevant dynamics. Because this channel operates on observations rather than actions, it is largely indifferent to the morphology and behavioral style that vary across embodiments. We hypothesize that task-relevant dynamics transfer across embodiments more readily than actions, so human data can shape the shared backbone through the world-model channel even when its action labels cannot. By decoupling supervision into "*how the world evolves*" and "*how each embodiment acts*", WAMs induce a more shareable representation and enable more effective scaling from *large-scale in-the-wild human data* than action-level co-training can achieve.

The promise of WAMs for human-robot co-training, however, hinges on a question that has not been systematically studied: ***what world representation enables effective transfer across embodiments under WAM co-training?*** Most WAMs predict pixels through a pretrained video VAE [8, 9, 10, 11] whose latent is optimized for photometric reconstruction and entangles motion with appearance, which we identify as the dominant failure mode of pixel-level WAM co-training. We argue that the world representation should satisfy three desiderata: *appearance abstraction*, *cross-embodiment consistency*, and *ego-motion factoring*. We then study two alternatives that satisfy them to different degrees: DINO features [12, 13], whose semantic prior abstracts appearance and aligns predictions across embodiments but remains spatially indexed on the image grid, and 3D motion flow [14, 15], which satisfies all three by construction through camera-frame-aligned geometric grounding.

We introduce **EgoWAM**, a framework for human-robot WAM co-training built upon a Heterogeneous Pretrained Transformer backbone [16, 7]. EgoWAM features a shared action head and a *swappable* world-model head trained jointly on human and robot data, isolating the effect of world representation under a matched backbone and data mixture. Evaluated on three real-world bimanual tasks spanning spatial, object, and scene generalization, our study makes three contributions:

- **A framework for controlled study of WAM co-training.** EgoWAM provides a single backbone with a swappable world-model head, enabling apples-to-apples comparison of world representations under identical action supervision and data mixtures.
- **Evidence that WAM co-training unlocks human-data scale.** Across in-domain and OOD evaluations, WAM co-training scales and transfers more consistently from large-scale natural egocentric human data than BC co-training, confirming that future-dynamics supervision is the missing channel where action-only supervision saturates.
- **World representation as the next critical axis.** Both DINO features and 3D motion flow substantially outperform the pixel-VAE baseline, with complementary strengths: DINO's semantic prior yields the strongest object and scene generalization, while 3D flow's geometric grounding yields the strongest spatial generalization and the highest overall scores.

2 Related Work

Robot Learning from Human Data. Human video is a scalable data source for robot manipulation [2, 1, 7]. One line of work treats it as a pretraining corpus for action-aware visual representations later transferred to downstream planning or policy [17, 18, 19, 20]. Beyond pretraining, prior work uses human data more directly for manipulation along two axes. The first retargets human demonstrations into a shared action space and co-trains an imitation policy [4, 5, 21, 22, 23, 24] or world

model [25, 26, 27] on both human and robot data; because of domain gaps across embodiments, this hinges on tight alignment in viewpoint, speed, and kinematics [6, 28, 29, 30, 31]. The second axis sidesteps the action decoder by extracting object or scene motion from human video and decoding actions from it at inference, especially with 2D point trajectory tracks [32, 33, 34, 35, 36, 37] and 3D flow motion fields [38, 39, 40, 41, 42]. EGOWAM departs from both: scene motion supervises the shared trunk during *end-to-end* training and is discarded at inference, transferring task-relevant dynamics without injecting inexecutable motions or adding test-time cost.

World Action Models for Robot Learning. A world model predicts how the environment evolves given an action [43, 44, 45, 46]; a World Action Model (WAM) couples such a predictor to a policy, grounding action prediction in task-relevant dynamics [26, 10, 11]. Most existing WAMs build on a pretrained video model to offer a spatio-temporal representation for action decoding [8, 9, 47, 48, 49], while a parallel line instead treats the video/world model as an interactive simulator for data generation or policy evaluation [50, 51, 52]. A central but unresolved question for generalizable WAMs is *what world representation* supports scaling from diverse data [53, 54, 55, 56, 57]. We study this question directly inside the WAM co-training framework, comparing three representations under a single training-as-representation-shaping setup [58, 10]: reconstruction-based pixel latents [59], semantics-based DINO features [12, 13, 60, 61], and camera-stabilized 3D motion flow [14, 15].

3 Human-Robot WAM Co-Training

3.1 Aligned Action Channel as a Strong Baseline

We first align the action channel as much as possible so BC co-training is a strong baseline, not a strawman [7]. We consider an egocentric human dataset $\mathcal{D}_H = \{(o_t^H, a_t^H)\}_{t=1}^{N_H}$ collected with Project Aria glasses [62], and a bimanual teleoperated robot dataset $\mathcal{D}_R = \{(o_t^R, a_t^R)\}_{t=1}^{N_R}$. Both share an egocentric RGB stream I^{ego} and the robot additionally has wrist-mounted views I^{wrist}.

We unify cross-embodiment actions into a 14-D end-effector space: per-arm 6-DoF $\mathrm{SE}(3)$ pose plus a 1-D gripper command. Robot actions $a_{t:t+k}^R \in \mathbb{R}^{k \times d_a}$ are computed from joint angles via forward kinematics and re-expressed in the static ego-camera frame. Human hand poses $p_{t+i}^H \in \mathrm{SE}(3)$ are natively expressed in the moving device frame with transform T_t^{device}; we re-express them in the *instantaneous* device frame at time t, $a_{t:t+k}^H = \left[(T_t^{\text{device}})^{-1} T_{t+i}^{\text{device}}\, p_{t+i}^H\right]_{i=1}^{k}$, factoring out head ego-motion so the same physical motion yields a comparable numerical trajectory across embodiments.

Two residual mismatches remain after coordinate alignment. *(i) Speed.* Humans act faster than teleoperated robots, so we use embodiment-specific windows spanning comparable progress—$T_H = 1\,\text{s}$, $T_R = 1.5\,\text{s}$—both discretized into k steps, yielding semantically aligned trajectories. We use k for resampled chunk length and $T \in \{T_H, T_R\}$ for the original-time horizon, with world-model targets s_{t+T} indexed in original time. *(ii) Workspace range.* Quantile normalization maps each action dimension's 1st and 99th percentiles to $[-1, 1]$, robust to hand-tracking outliers.

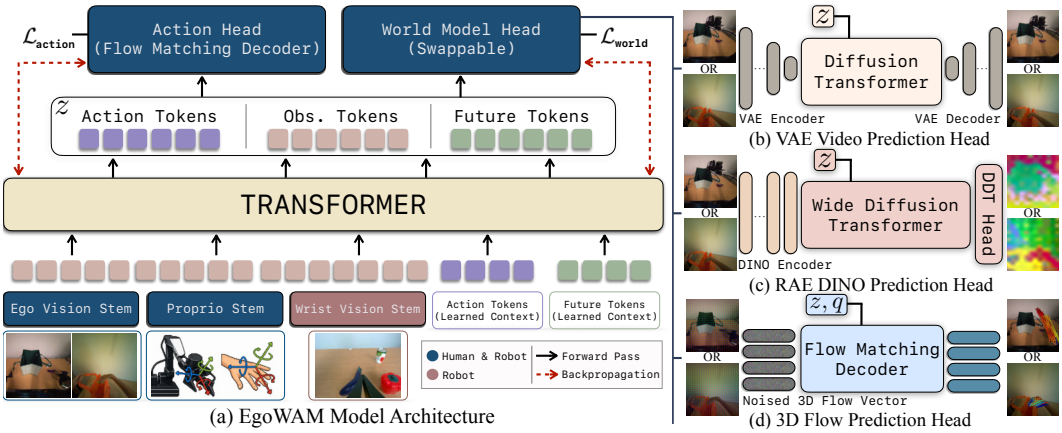
Given these aligned inputs, a shared encoder $f_\phi : \mathcal{O}_H \cup \mathcal{O}_R \to \mathcal{Z}$ and a shared action decoder $\pi_\theta(a \mid z)$ are trained end-to-end with the cross-embodiment BC objective

$$\mathcal{L}_{\text{BC-cotrain}}(\phi, \theta) = \sum_{\mathcal{D} \in \{\mathcal{D}_H, \mathcal{D}_R\}} \mathbb{E}_{(o,a)\sim\mathcal{D}}\, \mathcal{L}_{\text{BC}}(\pi_\theta(a \mid f_\phi(o)),\, a)\,, \tag{1}$$

realized as the sum of per-embodiment conditional flow-matching losses, $\mathcal{L}_{\text{BC-cotrain}} = \mathcal{L}_{\text{CFM}}^{\text{robot}} + \mathcal{L}_{\text{CFM}}^{\text{human}}$. This is the strongest action-aligned baseline our system supports; the remaining transfer gap motivates the world-model interface introduced next.

3.2 WAM as a World-Level Transfer Interface

To open a second supervision channel, we augment the BC policy with an auxiliary future-prediction head, yielding the World Action Model (WAM) family. The shared encoder f_ϕ maps the current observation to a latent $z_t = f_\phi(o_t)$, from which two parallel heads read out the modalities of interest: an action head $\pi_\theta(a \mid z)$ producing the chunk $a_{t:t+k} \in \mathbb{R}^{k \times d_a}$, and a world-model head $g_\psi(s \mid z)$



(a) EgoWAM Model Architecture (b) VAE Video Prediction Head (c) RAE DINO Prediction Head (d) 3D Flow Prediction Head

Figure 2: **Model Architecture.** **(a)** EgoWAM builds on a Heterogeneous Pretrained Transformer [16, 7] with modality-specific stems (ego vision, proprioception, wrist vision), learned action and future tokens, a flow-matching action head, and a swappable world-model head supplying dynamics supervision that carries human data where actions cannot. The head supports three world targets: **(b)** a VAE head decoding pixel latents, **(c)** an RAE DINO-feature head, and **(d)** a 3D-flow head denoising camera-stabilized motion flow.

producing a future state s_{t+T} at horizon T, *fused within the trunk*:

$$p_{\theta,\psi}\big(a_{t:t+k},\, s_{t+T} \,\big|\, o_t\big) = p_\psi\big(s_{t+T} \mid z_t\big)\, p_\theta\big(a_{t:t+k} \mid z_t\big), \qquad z_t = f_\phi(o_t), \tag{2}$$

with both heads trained jointly under $\mathcal{L}_{\text{WAM}} = \mathcal{L}_{\text{action}}(a_{t:t+k}) + \lambda\, \mathcal{L}_{\text{world}}(s_{t+T})$, where λ trades off action fidelity against future prediction supervision. The key property is that $\mathcal{L}_{\text{world}}$ supervises the *shared trunk* through dynamics produced by *both* embodiments, while $\mathcal{L}_{\text{action}}$ supervises through aligned actions we exhausted in Sec. 3.1. Human data can therefore shape the shared representation z_t through future-scene prediction even when its action labels do not transfer faithfully.

This reframing exposes the central question that prior WAM work has not systematically addressed: ***what should*** s_{t+T} ***be?*** The world target decides whether human and robot data converge to a shared representation or are pulled apart by embodiment-specific signal. We turn to this question next.

4 EgoWAM: A Controlled Study of World Representations

For WAM co-training to transfer from egocentric human data to robot manipulation, we posit three desiderata for the world representation:

- **(D1) Appearance abstraction.** Targets rewarding photometric reconstruction force the trunk to encode embodiment-specific appearance, crowding out the structure that governs task outcomes.
- **(D2) Cross-embodiment consistency.** Targets should represent the *effect* rather than the *agent*, so a human hand and a robot gripper producing similar physical change induce similar supervision.
- **(D3) Ego-motion factoring.** Image-coordinate targets conflate head rotation with scene change, giving the same event different supervision under moving human versus static robot cameras.

These desiderata define the axis EgoWAM studies; we instantiate three targets along it—pixel VAE, DINO features, and 3D flow. The overview of our model architecture is shown in Fig. 2. Further implementation details are provided in Appendix B.

4.1 Single-Backbone Architecture and Controlled Variables

EgoWAM builds on the Heterogeneous Pretrained Transformer (HPT) backbone [16, 7], where embodiment-specific stems tokenize each input into a shared latent space and a single transformer trunk operates on the unified token stream. This inductive bias extends naturally to WAMs: observation, action, and future-prediction tokens coexist in one stream and attend within the same trunk, exposing future dynamics and action prediction as two read-outs of a shared latent. We inherit the tokenizer and trunk and add a second bank of learnable *future tokens* and a *swappable* world-model head that consumes them (Fig. 2).

Heterogeneous Tokenizers. Each embodiment is tokenized through a shallow, embodiment-specific stem. A shared ego-vision stem encodes I_t^{ego} with a ResNet-18 encoder [63] or pretrained DINO encoder [12, 61]; a matching wrist-vision stem encodes I_t^{wrist} on robot batches. The proprioception stem encodes the end-effector pose with a per-embodiment MLP. Shared learnable *action* and *future* tokens then query the trunk, which routes them to the action and world-model heads.

Action Head. The action head π_θ is a multi-block transformer decoder trained with conditional flow matching [64, 7]. Given a clean chunk $a_{t:t+k}$ and noise $\epsilon \sim \mathcal{N}(0, I)$, we draw $\tau \sim \text{Beta}(1.5, 1.0)$ and form $a^\tau_{t:t+k} = (1-\tau)\epsilon + \tau\, a_{t:t+k}$ to initialize the action tokens, with τ embedded along the hidden dimension. Alternating self- and cross-attention blocks denoise the tokens while injecting trunk context, and a linear layer projects them into the action dimension.

Swappable World-Model Head. The world-model head g_ψ is conditioned on the trunk embeddings and predicts the target s_{t+T} at the embodiment-specific horizon T (Sec. 3.1). We defer the choices of world-model head and target to Sec. 4.2, where we vary the world representation while holding the trunk, action head, and data mixture fixed.

4.2 World-Model Target Instantiations

We instantiate three world targets spanning desiderata (D1)–(D3), each pairing a target s with a suitable head and trained under a shared linear path $s^\tau = (1-\tau)\epsilon + \tau s$, $\tau \in [0, 1]$, $\epsilon \sim \mathcal{N}(0, I)$.

Pixel VAE (Reconstruction Baseline). The Pixel-VAE variant predicts a future ego frame in the latent space of a pretrained video VAE [59] optimized for photometric reconstruction: $s = \text{VAE}(I^{\text{ego}}_{t+T})$. The head is a diffusion transformer (DiT) following the VACE-1.3B [65] architecture without text conditioning, initialized either from pretrained weights (**Pixel-PT**) or from scratch (**Pixel**), and trained to predict noise: $\mathcal{L}^{\text{VAE}}_{\text{world}} = \mathbb{E}\big\|\epsilon - \epsilon_\psi(s^\tau, \tau, f_\phi(o))\big\|^2$. This target violates all three desiderata and serves as our reconstruction-level baseline.

DINO Features (Semantic Abstraction). The target is DINO [12] patch features of the ego frame at $t+T$: $s = \text{DINO}(I^{\text{ego}}_{t+T})$, replacing photometric fidelity with prediction in a semantically structured space. The head follows the wide-DDT design of RAE [13], which widens the denoiser to match the high channel count of semantic latents and decouples noise-prediction from feature-conditioning blocks. The objective remains noise prediction: $\mathcal{L}^{\text{RAE}}_{\text{world}} = \mathbb{E}\big\|\epsilon - \epsilon_\psi(s^\tau, \tau, f_\phi(o))\big\|^2$. DINO features remain spatially indexed in image coordinates, so head ego-motion (D3) is only partially mitigated.

3D Flow (Spatial Abstraction). The target is a dense 3D motion field over $[t, t+T]$, expressed in the camera-stabilized frame at time t: $s = F_{[t,t+T]}$. Raw 3D point displacement on egocentric video is dominated by ego-motion: stationary objects induce large apparent flow whenever the wearer turns their head. We address this by feeding the pretrained 3D point tracker [14, 15] with Aria VIO camera poses [62], so the returned point positions X_t, X_{t+T} share a consistent world frame. We then map the future position back to the camera frame at t: $\tilde{X}_{t+T} = (T^{\text{cam}}_t)^{-1}\, T^{\text{cam}}_{t+T}\, X_{t+T}$, and define the flow target as $s = F_{[t,t+T]} = \tilde{X}_{t+T} - X_t$. After stabilization, static background yields near-zero flow while manipulated objects retain motion proportional to physical displacement, abstracting dynamics from both appearance and viewpoint. The head is an alternating self/cross-attention decoder that takes query points q uniformly sampled from the current ego frame as additional conditioning, and regresses velocity at those points: $\mathcal{L}^{\text{Flow}}_{\text{world}} = \mathbb{E}\big\|u_\psi(s^\tau, \tau, f_\phi(o), q) - (s_q - \epsilon_q)\big\|^2$. This target satisfies all three desiderata by construction and is the most spatially grounded of the three.

4.3 Joint Training and Action-Only Inference

Joint Training. EGOWAM is trained end-to-end under

$$\mathcal{L}_{\text{EGOWAM}} = \underbrace{\mathcal{L}^{\text{robot}}_{\text{action}} + \mathcal{L}^{\text{human}}_{\text{action}}}_{\mathcal{L}_{\text{action}}} + \lambda \underbrace{\big(\mathcal{L}^{\text{robot}}_{\text{world}} + \mathcal{L}^{\text{human}}_{\text{world}}\big)}_{\mathcal{L}_{\text{world}}}, \tag{3}$$

with $\lambda = 1$. Each step draws a mini-batch from $\mathcal{D}_R$ and $\mathcal{D}_H$; both pass through the shared tokenizers and trunk, after which the action head yields $\mathcal{L}_{\text{action}}$ (Eq. 1) and the world-model head yields $\mathcal{L}_{\text{world}}$

(Sec. 4.2). Both losses supervise the shared trunk f_ϕ: when human action labels transfer weakly, world prediction compensates and shapes the trunk representation.

Action-Only Inference. Recent work [58, 10] shows that WAMs benefit from *training-time representation shaping* rather than test-time imagination. We therefore drop the world-model head at inference and unroll only the action head to decode $a_{t:t+k}$ from the trunk embeddings, matching the latency of a same-size BC policy (30 Hz) while retaining the cross-embodiment representation learned under joint supervision. This decoupling is central to our positioning: EGOWAM is an *instrument for studying which world representation transfers*, with findings deployable at BC cost.

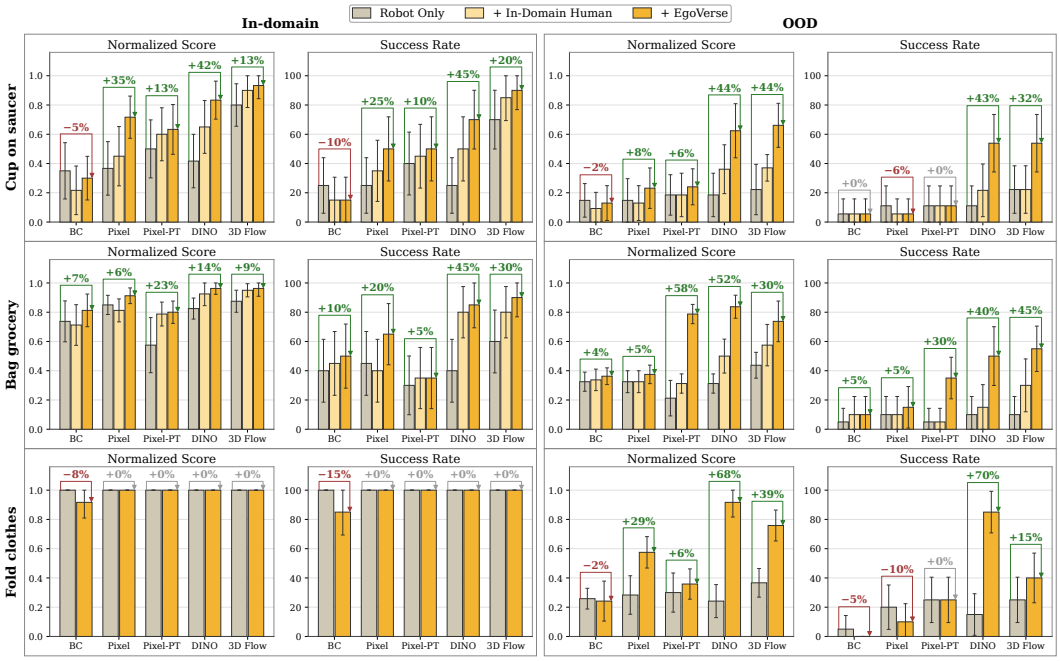


Figure 3: **Quantitative Comparison on Real-World Rollouts.** Normalized score and success rate across three bimanual tasks under ID and OOD evaluation, comparing BC against four WAM variants. WAM co-training consistently outperforms BC: human data often degrades BC yet yields large WAM gains. Pixel transfers weakly, while DINO drives the strongest OOD generalization and 3D Flow the largest ID spatial gains. Error bars indicate 95% finite-sample-valid confidence intervals with Type-I error control for miscoverage [66].

5 Experiments

We present a systematic study of EGOWAM for human-robot co-training, organized around three main questions: **Q1**: Does WAM co-training scale and transfer from large-scale in-the-wild human data better than BC co-training? **Q2**: Which world representation best enables transfer from human data to robot manipulation? **Q3**: How robust is each paradigm to action-misaligned human data?

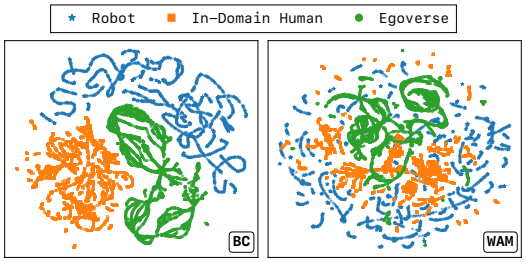


Figure 4: **UMAP of Trunk Embeddings on Cup-on-Saucer.** BC separates human-robot embeddings; WAM aligns them in a shared latent.

We provide additional experimental results and analysis in Appendix A and robot-to-robot transfer simulation results in Appendix D.

5.1 Experimental Setup

Hardware Setup. Our bimanual robot platform has two upright-mounted 6-DoF ARX5 arms with parallel-jaw grippers, head-mounted Project Aria glasses [62] for egocentric RGB shared with human demonstrations, and two wrist-mounted Intel RealSense D405 cameras. Robot actions are

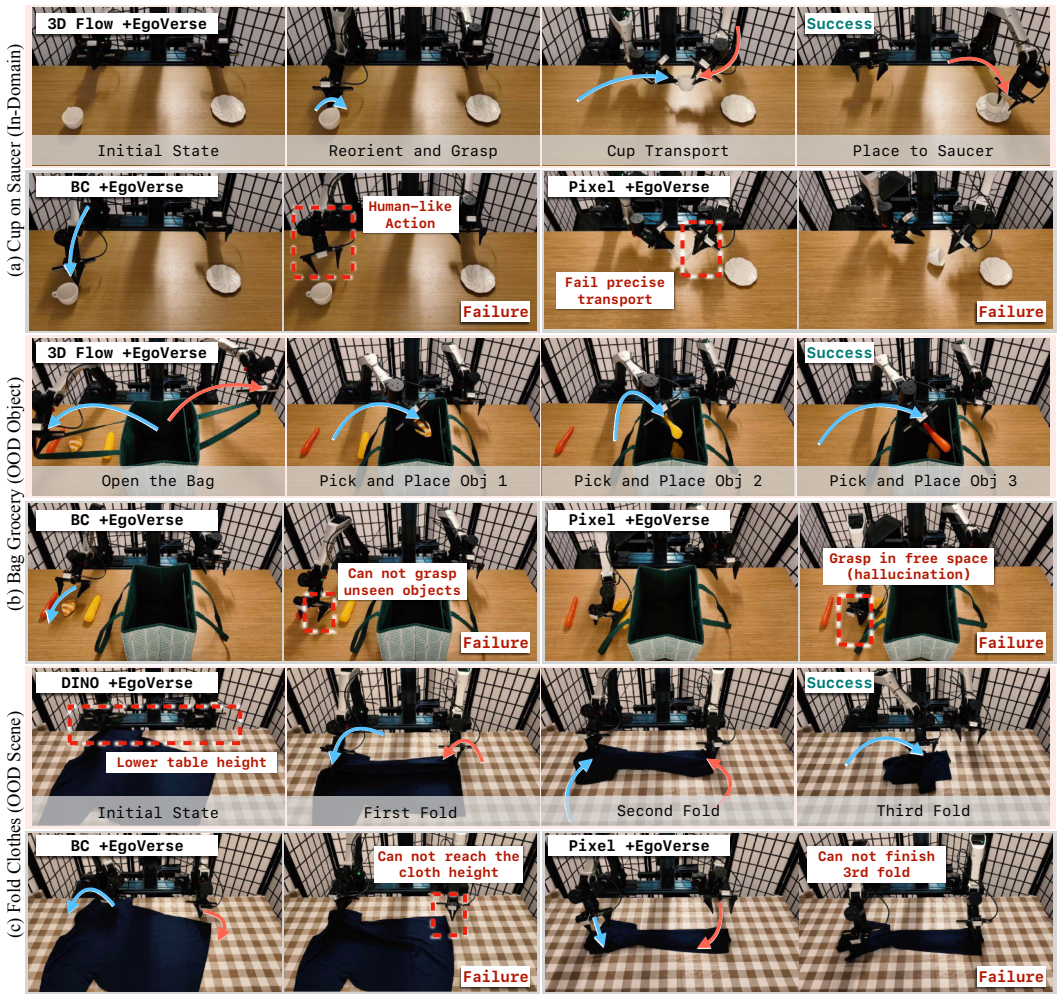


Figure 5: **Qualitative Comparison on Real-World Rollouts.** WAM variants (3D Flow, DINO) are compared against BC and Pixel baselines under *EgoVerse co-training*. **(a)** BC produces human-like motion, Pixel fails at precise cup transport; **(b)** BC cannot grasp unseen objects, Pixel hallucinates a free-space grasp; **(c)** BC overfits and fails to reach the cloth, Pixel's geometric confusion leaves the third fold incomplete.

per-arm 6-DoF end-effector poses with gripper state ($a^R_{t:t+k} \in \mathbb{R}^{k\times 14}$); human and robot data are unified into the camera-centered $\mathrm{SE}(3)$ action space and quantile-normalized (Sec. 3.1).

Tasks. We evaluate on three bimanual tasks from the EgoVerse flagship set [7], spanning precise rigid-object manipulation, deformable manipulation, and long-horizon sequencing: (1) **cup-on-saucer**: Reorient a cup from a randomized pose and place it upright on a saucer at a randomized position; (2) **fold-clothes**: Three-fold a T-shirt from random initial configurations; (3) **bag-grocery**: Open a grocery bag and load three items into it from randomized positions.

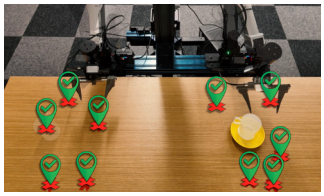


Figure 6: **Spatial Gains for 3D Flow.** 3D Flow succeeds (green) where BC fails (red) at cup positions across the workspace.

Data. We collect 300–360 robot demonstrations per task via Meta Quest 3, with randomized placements, orientations, and 4–8 object combinations. Human data spans two regimes varying in *scale* and *observation alignment*: (1) **In-Domain Human** (**1:1** with robot data): same scenes and objects as the robot, but unmatched viewpoint and behavior; (2) **EgoVerse** (∼**10:1**): the full EgoVerse-A flagship split per task [7], with diverse scenes, objects, and demonstrators and no deliberate scene and object alignment.

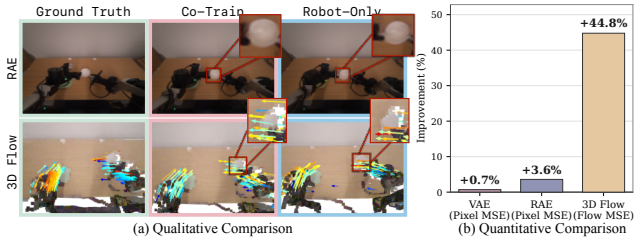
Evaluation Protocol. Each method is evaluated on **ID** (20 rollouts: seen objects and scene, randomized positions and orientations) and **OOD** (20 rollouts: 10 unseen objects in the training scene,

10 seen objects in novel scenes with varied backgrounds and table heights). We report normalized sub-task score and success rate over **1800** total real-world rollouts.

Further details on the experimental setup are provided in Appendix C.

5.2 Core Results and Findings

(Q1) WAM vs. BC Co-Training under Natural Human Data. When the robot and human action patterns are not well aligned—for example, the human hand tends to hold the cup sideways while the gripper grasps it from the front (Fig. 1)—BC degrades by reproducing these *inexecutable human-like actions* (Fig. 5), whereas WAM consistently turns the same data into reliable gains. Moreover, even when large amounts of human data are introduced, BC tends to *overfit* to the robot data alone and fails to benefit from human data for generalization. Fig. 5 shows that on the fold-clothes task, BC+EgoVerse still overfits to the robot data and cannot adapt to novel scenes with lower table heights, whereas the DINO-based WAM leverages in-the-wild human data to achieve better generalization to new objects and scenes. Fig. 4 traces this to representation: BC isolates human and robot (and even human–human) embeddings under unaligned actions, whereas WAM aligns them into a shared space through additional task-relevant dynamics supervision.



(a) Qualitative Comparison (b) Quantitative Comparison

Figure 7: **World Prediction Comparison.** **(a)** Co-training improves RAE and 3D Flow predictions, with better *object shape and motion*; **(b)** 3D Flow's gains far exceed Pixel and DINO.

(Q2) World Representation Choice for Cross-Embodiment Transfer. The world representation is the next critical axis. Fig. 3 shows that pixel-based prediction transfers weakly: reconstructing raw appearance entangles embodiment- and scene-specific detail that does not transfer, leaving the hallucination and geometric confusion in Fig. 5: a free-space grasp on bag-grocery and an unfinished final fold on fold-clothes. Abstraction fixes this in two complementary ways: the DINO-based WAM predicts semantic features and yields the strongest OOD generalization to unseen objects and scenes (Fig. 3), whereas the 3D-flow WAM predicts camera-stabilized motion and delivers the largest in-domain spatial gains, succeeding at precise cup placement across the workspace (Fig. 6). Fig. 7 grounds this contrast: pixel (VAE) prediction barely changes, the semantic (RAE) target refines *object shape* from human data but only moderately, and the 3D-flow target benefits most, recovering *object motion* that robot-only prediction leaves static.

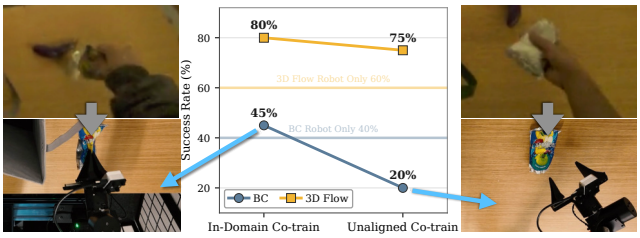


Figure 8: **Ablation of Unaligned Human Data.** Unaligned human data collapses BC below robot-only; WAM stays robust.

(Q3) Ablation: The Role of Unaligned Human Data. We further study the hypothesis that unaligned human data degrades BC. Among the three tasks, only bag-grocery benefits from action-level co-training (Fig. 3), because its pick-and-place motions are naturally aligned between human and robot (Fig. 8, left). As a counterfactual, we collect unaligned human demonstrations in which the human picks objects in an unusual manner that maps to inexecutable robot actions (Fig. 8, right). Under this data, BC collapses below the robot-only baseline, whereas the 3D-flow WAM stays robust and still surpasses robot-only. The results on ablation of aligned human data are shown in App. A.1.

6 Conclusion

We introduce EgoWAM, a human-robot WAM co-training framework studying what *world representation* enables cross-embodiment transfer. It scales with *in-the-wild human data* where action-

only BC stalls, and the world representation proves critical: pixels transfer weakly, DINO drives object and scene generalization, and 3D flow grounds in-domain spatial gains. This yields a recipe for further study: a target that abstracts appearance, keeps effects consistent across embodiments, and factors out ego-motion, positioning human data as a flywheel for scalable robot learning. We offer this study not as a definitive answer but as a point of departure, turning "use human data" into a concrete design axis for scaling robot manipulation toward open-world generalization.

7 Limitations

Three limitations open future directions. **(1) Motion generalization.** Our gains stay at the context level; learning novel motion primitive / skill from human data (e.g., folding shorts from a T-shirt model) remains out of reach, and a more unified action representation is left to future study. **(2) Multi-task scaling.** We isolate world representation with one policy per task; multi-task co-training on large-scale in-the-wild human data is promising to explore. **(3) Open world representation.** Our study is a starting point, showing DINO and 3D-flow beat pixels for cross-embodiment transfer; the "best" world representation in scaling robot learning remains a valuable open research question.

Acknowledgments

This work was supported in part by the Toyota Research Institute through the TRI University 3.0 program. We thank the members of the Robot Learning and Reasoning Lab (RL2) for helpful discussions and hardware support.

References

[1] R. Hoque, P. Huang, D. J. Yoon, M. Sivapurapu, and J. Zhang. EgoDex: Learning Dexterous Manipulation from Large-Scale Egocentric Video. *arXiv preprint arXiv:2505.11709*, 2026.

[2] K. Grauman, A. Westbury, L. Torresani, K. Kitani, J. Malik, T. Afouras, K. Ashutosh, V. Baiyya, S. Bansal, B. Boote, E. Byrne, Z. Chavis, J. Chen, F. Cheng, F.-J. Chu, S. Crane, A. Dasgupta, J. Dong, M. Escobar, C. Forigua, A. Gebreselasie, S. Haresh, J. Huang, M. M. Islam, S. Jain, R. Khirodkar, D. Kukreja, K. J. Liang, J.-W. Liu, S. Majumder, Y. Mao, M. Martin, E. Mavroudi, T. Nagarajan, F. Ragusa, S. K. Ramakrishnan, L. Seminara, A. Somayazulu, Y. Song, S. Su, Z. Xue, E. Zhang, J. Zhang, A. Castillo, C. Chen, X. Fu, R. Furuta, C. Gonzalez, P. Gupta, J. Hu, Y. Huang, Y. Huang, W. Khoo, A. Kumar, R. Kuo, S. Lakhavani, M. Liu, M. Luo, Z. Luo, B. Meredith, A. Miller, O. Oguntola, X. Pan, P. Peng, S. Pramanick, M. Ramazanova, F. Ryan, W. Shan, K. Somasundaram, C. Song, A. Southerland, M. Tateno, H. Wang, Y. Wang, T. Yagi, M. Yan, X. Yang, Z. Yu, S. C. Zha, C. Zhao, Z. Zhao, Z. Zhu, J. Zhuo, P. Arbelaez, G. Bertasius, D. Crandall, D. Damen, J. Engel, G. M. Farinella, A. Furnari, B. Ghanem, J. Hoffman, C. V. Jawahar, R. Newcombe, H. S. Park, J. M. Rehg, Y. Sato, M. Savva, J. Shi, M. Z. Shou, and M. Wray. Ego-Exo4D: Understanding Skilled Human Activity from First- and Third-Person Perspectives. *arXiv preprint arXiv:2311.18259*, 2024.

[3] S. Kareer, K. Pertsch, J. Darpinian, J. Hoffman, D. Xu, S. Levine, C. Finn, and S. Nair. Emergence of Human to Robot Transfer in Vision-Language-Action Models. *arXiv preprint arXiv:2512.22414*, 2025.

[4] S. Kareer, D. Patel, R. Punamiya, P. Mathur, S. Cheng, C. Wang, J. Hoffman, and D. Xu. EgoMimic: Scaling Imitation Learning via Egocentric Video. *arXiv preprint arXiv:2410.24221*, 2024.

[5] R.-Z. Qiu, S. Yang, X. Cheng, C. Chawla, J. Li, T. He, G. Yan, D. J. Yoon, R. Hoque, L. Paulsen, G. Yang, J. Zhang, S. Yi, G. Shi, and X. Wang. Humanoid policy ˜ human policy. *arXiv preprint arXiv:2503.13441*, 2025.

[6] R. Zheng, D. Niu, Y. Xie, J. Wang, M. Xu, Y. Jiang, F. Castañeda, F. Hu, Y. L. Tan, L. Fu, T. Darrell, F. Huang, Y. Zhu, D. Xu, and L. Fan. EgoScale: Scaling Dexterous Manipulation with Diverse Egocentric Human Data. *arXiv preprint arXiv:2602.16710*, 2026.

[7] R. Punamiya, S. Kareer, Z. Liu, J. Citron, R.-Z. Qiu, X. Cai, A. Gavryushin, J. Chen, D. Liconti, L. Y. Zhu, P. Aphiwetsa, B. Li, A. Cheluva, P. Kuppili, Y. Liu, D. Patel, A. Gao, H.-Y. Chung, R. Co, R. Zbizika, J. Liu, X. Xu, H. Xiong, G. Chen, S. Oliani, C. Yang, X. Wang, J. Fort, R. Newcombe, J. Gao, J. Chong, G. Matsuda, A. Doriwala, M. Pollefeys, R. Katzschmann, X. Wang, S. Song, J. Hoffman, and D. Xu. EgoVerse: An Egocentric Human Dataset for Robot Learning from Around the World. *arXiv preprint arXiv:2604.07607*, 2026.

[8] S. Ye, Y. Ge, K. Zheng, S. Gao, S. Yu, G. Kurian, S. Indupuru, Y. L. Tan, C. Zhu, J. Xiang, A. Malik, K. Lee, W. Liang, N. Ranawaka, J. Gu, Y. Xu, G. Wang, F. Hu, A. Narayan, J. Bjorck, J. Wang, G. Kim, D. Niu, R. Zheng, Y. Xie, J. Wu, Q. Wang, R. Julian, D. Xu, Y. Du, Y. Chebotar, S. Reed, J. Kautz, Y. Zhu, L. J. Fan, and J. Jang. World Action Models are Zero-shot Policies. *arXiv preprint arXiv:2602.15922*, 2026.

[9] M. J. Kim, Y. Gao, T.-Y. Lin, Y.-C. Lin, Y. Ge, G. Lam, P. Liang, S. Song, M.-Y. Liu, C. Finn, and J. Gu. Cosmos Policy: Fine-Tuning Video Models for Visuomotor Control and Planning. *arXiv preprint arXiv:2601.16163*, 2026.

[10] S. Li, Y. Gao, D. Sadigh, and S. Song. Unified Video Action Model. *arXiv preprint arXiv:2503.00200*, 2025.

[11] C. Zhu, R. Yu, S. Feng, B. Burchfiel, P. Shah, and A. Gupta. Unified World Models: Coupling Video and Action Diffusion for Pretraining on Large Robotic Datasets. *arXiv preprint arXiv:2504.02792*, 2025.

[12] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, R. Howes, P.-Y. Huang, H. Xu, V. Sharma, S.-W. Li, W. Galuba, M. Rabbat, M. Assran, N. Ballas, G. Synnaeve, I. Misra, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.

[13] B. Zheng, N. Ma, S. Tong, and S. Xie. Diffusion Transformers with Representation Autoencoders. *arXiv preprint arXiv:2510.11690*, 2025.

[14] Y. Xiao, J. Wang, N. Xue, N. Karaev, Y. Makarov, B. Kang, X. Zhu, H. Bao, Y. Shen, and X. Zhou. SpatialTrackerV2: 3D Point Tracking Made Easy. *arXiv preprint arXiv:2507.12462*, 2025.

[15] J. Lu, J. Xu, W. Hu, R. Zhu, C. Zhao, S.-K. Yeung, Y. Shan, and Y. Liu. Track4world: Feedforward world-centric dense 3d tracking of all pixels. *arXiv preprint arXiv:2603.02573*, 2026.

[16] L. Wang, X. Chen, J. Zhao, and K. He. Scaling Proprioceptive-Visual Learning with Heterogeneous Pre-trained Transformers. *arXiv preprint arXiv:2409.20537*, 2024.

[17] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta. R3M: A Universal Visual Representation for Robot Manipulation. *arXiv preprint arXiv:2203.12601*, 2022.

[18] M. Assran, A. Bardes, D. Fan, Q. Garrido, R. Howes, Mojtaba, Komeili, M. Muckley, A. Rizvi, C. Roberts, K. Sinha, A. Zholus, S. Arnaud, A. Gejji, A. Martin, F. R. Hogan, D. Dugas, P. Bojanowski, V. Khalidov, P. Labatut, F. Massa, M. Szafraniec, K. Krishnakumar, Y. Li, X. Ma, S. Chandar, F. Meier, Y. LeCun, M. Rabbat, and N. Ballas. V-jepa 2: Self-supervised video models enable understanding, prediction and planning, 2025. URL https://arxiv.org/abs/2506.09985.

[19] D. Niu, Y. Sharma, H. Xue, G. Biamby, J. Zhang, Z. Ji, T. Darrell, and R. Herzig. Pre-training auto-regressive robotic models with 4d representations. *arXiv preprint arXiv:2502.13142*, 2025.

[20] Y. Bai, D. Tran, A. Bar, Y. LeCun, T. Darrell, and J. Malik. Whole-body conditioned egocentric video prediction, 2025. URL https://arxiv.org/abs/2506.21552.

[21] R. Yang, Q. Yu, Y. Wu, R. Yan, B. Li, A.-C. Cheng, X. Zou, Y. Fang, X. Cheng, R.-Z. Qiu, H. Yin, S. Liu, S. Han, Y. Lu, and X. Wang. EgoVLA: Learning Vision-Language-Action Models from Egocentric Human Videos. *arXiv preprint arXiv:2507.12440*, 2025.

[22] H. Bi, L. Wu, T. Lin, H. Tan, Z. Su, H. Su, and J. Zhu. H-RDT: Human Manipulation Enhanced Bimanual Robotic Manipulation. *arXiv preprint arXiv:2507.23523*, 2025.

[23] C. Yuan, R. Zhou, M. Liu, Y. Hu, S. Wang, L. Yi, C. Wen, S. Zhang, and Y. Gao. MotionTrans: Human VR Data Enable Motion-Level Learning for Robotic Manipulation Policies. *arXiv preprint arXiv:2509.17759*, 2025.

[24] Q. Li, Y. Deng, Y. Liang, L. Luo, L. Zhou, C. Yao, L. Zeng, Z. Feng, H. Liang, S. Xu, Y. Zhang, X. Chen, H. Chen, L. Sun, D. Chen, J. Yang, and B. Guo. Scalable Vision-Language-Action Model Pretraining for Robotic Manipulation with Real-Life Human Activity Videos. *arXiv preprint arXiv:2510.21571*, 2025.

[25] R. G. Goswami, A. Bar, D. Fan, T.-Y. Yang, G. Zhou, P. Krishnamurthy, M. Rabbat, F. Khorrami, and Y. LeCun. World models for learning dexterous hand-object interactions from human videos, 2026. URL https://arxiv.org/abs/2512.13644.

[26] B. Hou, G. Li, J. Jia, T. An, X. Guo, S. Leng, H. Geng, Y. Ze, T. Harada, P. Torr, O. Mees, M. Pollefeys, Z. Liu, J. Wu, P. Abbeel, J. Malik, Y. Du, and J. Yang. World model for robot learning: A comprehensive survey. 2026.

[27] R. Mendonca, S. Bahl, and D. Pathak. Structured World Models from Human Videos. *arXiv preprint arXiv:2308.10901*, 2023.

[28] G. Li, Y. Lyu, Z. Liu, C. Hou, J. Zhang, and S. Zhang. H2R: A Human-to-Robot Data Augmentation for Robot Pre-training from Videos. *arXiv preprint arXiv:2505.11920*, 2025.

[29] Y. Liu, W. C. Shin, Y. Han, Z. Chen, H. Ravichandar, and D. Xu. ImMimic: Cross-Domain Imitation from Human Videos via Mapping and Interpolation. *arXiv preprint arXiv:2509.10952*, 2025.

[30] R. Punamiya, D. Patel, P. Aphiwetsa, P. Kuppili, L. Y. Zhu, S. Kareer, J. Hoffman, and D. Xu. EgoBridge: Domain Adaptation for Generalizable Imitation from Egocentric Human Data. In *Advances in Neural Information Processing Systems*, 2025.

[31] X. Cai, R.-Z. Qiu, G. Chen, L. Wei, I. Liu, T. Huang, X. Cheng, and X. Wang. In-N-On: Scaling Egocentric Manipulation with in-the-wild and on-task Data. *arXiv preprint arXiv:2511.15704*, 2025.

[32] C. Wen, X. Lin, J. So, K. Chen, Q. Dou, Y. Gao, and P. Abbeel. Any-point Trajectory Modeling for Policy Learning. *arXiv preprint arXiv:2401.00025*, 2024.

[33] M. Xu, Z. Xu, Y. Xu, C. Chi, G. Wetzstein, M. Veloso, and S. Song. Flow as the Cross-Domain Manipulation Interface. *arXiv preprint arXiv:2407.15208*, 2024.

[34] J. Ren, P. Sundaresan, D. Sadigh, S. Choudhury, and J. Bohg. Motion Tracks: A Unified Representation for Human-Robot Transfer in Few-Shot Imitation Learning. *arXiv preprint arXiv:2501.06994*, 2025.

[35] S. Haldar and L. Pinto. Point Policy: Unifying Observations and Actions with Key Points for Robot Manipulation. *arXiv preprint arXiv:2502.20391*, 2025.

[36] S. Lee, Y. Jung, I. Chun, Y.-C. Lee, Z. Cai, H. Huang, A. Talreja, T. D. Dao, Y. Liang, J.-B. Huang, and F. Huang. TraceGen: World Modeling in 3D Trace Space Enables Learning from Cross-Embodiment Videos. *arXiv preprint arXiv:2511.21690*, 2025.

[37] J. A. Collins, L. Cheng, K. Aneja, A. Wilcox, B. Joffe, and A. Garg. Amplify: Actionless motion priors for robot learning from videos, 2025. URL https://arxiv.org/abs/2506.14198.

[38] A. Hung, B. P. Duisterhof, and J. Ichnowski. 3PoinTr: 3D Point Tracks for Robot Manipulation Pretraining from Casual Videos. *arXiv preprint arXiv:2603.08485*, 2026.

[39] H. Li, L. Sun, Y. Hu, D. Ta, J. Barry, G. Konidaris, and J. Fu. NovaFlow: Zero-Shot Manipulation via Actionable Flow from Generated Videos. *arXiv preprint arXiv:2510.08568*, 2025.

[40] Z.-H. Yin, S. Yang, and P. Abbeel. Object-centric 3D Motion Field for Robot Learning from Human Videos. *arXiv preprint arXiv:2506.04227*, 2025.

[41] D. Cho, Y. Jang, D. Xu, and S. Ha. EgoAVFlow: Robot Policy Learning with Active Vision from Human Egocentric Videos via 3D Flow. *arXiv preprint arXiv:2602.22461*, 2026.

[42] K. Dharmarajan, W. Huang, J. Wu, L. Fei-Fei, and R. Zhang. Dream2flow: Bridging video generation and open-world manipulation with 3d object flow. *arXiv preprint arXiv:2512.24766*, 2025.

[43] D. Ha and J. Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

[44] D. Hafner, W. Yan, and T. Lillicrap. Training agents inside of scalable world models. *arXiv preprint arXiv:2509.24527*, 2025.

[45] B. Ai, S. Tian, H. Shi, Y. Wang, T. Pfaff, C. Tan, H. I. Christensen, H. Su, J. Wu, and Y. Li. A review of learning-based dynamics models for robotic manipulation. *Science Robotics*, 10 (106):eadt1497, 2025.

[46] A. Bar, G. Zhou, D. Tran, T. Darrell, and Y. LeCun. Navigation world models, 2024. URL https://arxiv.org/abs/2412.03572.

[47] J. Pai, L. Achenbach, V. Montesinos, B. Forrai, O. Mees, and E. Nava. mimic-video: Video-Action Models for Generalizable Robot Control Beyond VLAs. *arXiv preprint arXiv:2512.15692*, 2025.

[48] T. Ma, J. Zheng, Z. Wang, C. Jiang, A. Cui, J. Liang, and S. Yang. DiT4DiT: Jointly Modeling Video Dynamics and Actions for Generalizable Robot Control. *arXiv preprint arXiv:2603.10448*, 2026.

[49] B. Chen, T. Zhang, H. Geng, K. Song, C. Zhang, P. Li, W. T. Freeman, J. Malik, P. Abbeel, R. Tedrake, V. Sitzmann, and Y. Du. Large Video Planner Enables Generalizable Robot Control. *arXiv preprint arXiv:2512.15840*, 2025.

[50] S. Gao, W. Liang, K. Zheng, A. Malik, S. Ye, S. Yu, W.-C. Tseng, Y. Dong, K. Mo, C.-H. Lin, Q. Ma, S. Nah, L. Magne, J. Xiang, Y. Xie, R. Zheng, D. Niu, Y. L. Tan, K. R. Zentner, G. Kurian, S. Indupuru, P. Jannaty, J. Gu, J. Zhang, J. Malik, P. Abbeel, M.-Y. Liu, Y. Zhu, J. Jang, and L. J. Fan. DreamDojo: A Generalist Robot World Model from Large-Scale Human Videos. *arXiv preprint arXiv:2602.06949*, 2026.

[51] Y. Wang, R. Syed, F. Wu, M. Zhang, A. Onol, J. Barreiros, H. Nayyeri, T. Dear, H. Zhang, and Y. Li. Interactive World Simulator for Robot Policy Training and Evaluation. *arXiv preprint arXiv:2603.08546*, 2026.

[52] K. Zhang, S. Sha, H. Jiang, M. Loper, H. Song, G. Cai, Z. Xu, X. Hu, C. Zheng, and Y. Li. Real-to-sim robot policy evaluation with gaussian splatting simulation of soft-body interactions. *arXiv preprint arXiv:2511.04665*, 2025.

[53] G. Zhou, H. Pan, Y. LeCun, and L. Pinto. DINO-WM: World Models on Pre-trained Visual Features enable Zero-shot Planning. *arXiv preprint arXiv:2411.04983*, 2025.

[54] Nilaksh, S. Jha, A. Zholus, and S. Chandar. Reconstruction or semantics? what makes a latent space useful for robotic world models. 2026. URL https://arxiv.org/abs/2605.06388.

[55] K. Zhang, B. Li, K. Hauser, and Y. Li. Adaptigraph: Material-adaptive graph-based neural dynamics for robotic manipulation. In *Proceedings of Robotics: Science and Systems (RSS)*, 2024.

[56] K. Zhang, B. Li, K. Hauser, and Y. Li. Particle-grid neural dynamics for learning deformable object models from rgb-d videos. In *Proceedings of Robotics: Science and Systems (RSS)*, 2025.

[57] W. Huang, Y.-W. Chao, A. Mousavian, M.-Y. Liu, D. Fox, K. Mo, and F.-F. Li. Point-world: Scaling 3d world models for in-the-wild robotic manipulation. *arXiv preprint arXiv:2601.03782*, 2026.

[58] T. Yuan, Z. Dong, Y. Liu, and H. Zhao. Fast-WAM: Do World Action Models Need Test-time Future Imagination? *arXiv preprint arXiv:2603.16666*, 2026.

[59] T. Wan, A. Wang, B. Ai, B. Wen, C. Mao, C.-W. Xie, D. Chen, F. Yu, H. Zhao, J. Yang, J. Zeng, J. Wang, J. Zhang, J. Zhou, J. Wang, J. Chen, K. Zhu, K. Zhao, K. Yan, L. Huang, M. Feng, N. Zhang, P. Li, P. Wu, R. Chu, R. Feng, S. Zhang, S. Sun, T. Fang, T. Wang, T. Gui, T. Weng, T. Shen, W. Lin, W. Wang, W. Wang, W. Zhou, W. Wang, W. Shen, W. Yu, X. Shi, X. Huang, X. Xu, Y. Kou, Y. Lv, Y. Li, Y. Liu, Y. Wang, Y. Zhang, Y. Huang, Y. Li, Y. Wu, Y. Liu, Y. Pan, Y. Zheng, Y. Hong, Y. Shi, Y. Feng, Z. Jiang, Z. Han, Z.-F. Wu, and Z. Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

[60] J. Singh, B. Zheng, Z. Wu, R. Zhang, E. Shechtman, and S. Xie. Improved baselines with representation autoencoders. *arXiv preprint arXiv:2605.18324*, 2026.

[61] O. Siméoni, H. V. Vo, M. Seitzer, F. Baldassarre, M. Oquab, C. Jose, V. Khalidov, M. Szafraniec, S. Yi, M. Ramamonjisoa, F. Massa, D. Haziza, L. Wehrstedt, J. Wang, T. Darcet, T. Moutakanni, L. Sentana, C. Roberts, A. Vedaldi, J. Tolan, J. Brandt, C. Couprie, J. Mairal, H. Jégou, P. Labatut, and P. Bojanowski. DINOv3, 2025. URL https://arxiv.org/abs/2508.10104.

[62] J. Engel, K. Somasundaram, M. Goesele, A. Sun, A. Gamino, A. Turner, A. Talattof, A. Yuan, B. Souti, B. Meredith, C. Peng, C. Sweeney, C. Wilson, D. Barnes, D. DeTone, D. Caruso, D. Valleroy, D. Ginjupalli, D. Frost, E. Miller, E. Mueggler, E. Oleinik, F. Zhang, G. Somasundaram, G. Solaira, H. Lanaras, H. Howard-Jenkins, H. Tang, H. J. Kim, J. Rivera, J. Luo, J. Dong, J. Straub, K. Bailey, K. Eckenhoff, L. Ma, L. Pesqueira, M. Schwesinger, M. Monge, N. Yang, N. Charron, N. Raina, O. Parkhi, P. Borschowa, P. Moulon, P. Gupta, R. Mur-Artal, R. Pennington, S. Kulkarni, S. Miglani, S. Gondi, S. Solanki, S. Diener, S. Cheng, S. Green, S. Saarinen, S. Patra, T. Mourikis, T. Whelan, T. Singh, V. Balntas, V. Baiyya, W. Dreewes, X. Pan, Y. Lou, Y. Zhao, Y. Mansour, Y. Zou, Z. Lv, Z. Wang, M. Yan, C. Ren, R. D. Nardi, and R. Newcombe. Project aria: A new tool for egocentric multi-modal ai research, 2023. URL https://arxiv.org/abs/2308.13561.

[63] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[64] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023.

[65] Z. Jiang, Z. Han, C. Mao, J. Zhang, Y. Pan, and Y. Liu. Vace: All-in-one video creation and editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

[66] J. A. Vincent, H. Nishimura, M. Itkina, P. Shah, M. Schwager, and T. Kollar. How generalizable is my behavior cloning policy? a statistical approach to trustworthy performance evaluation. *IEEE Robotics and Automation Letters*, 9(10):8619–8626, 2024. doi: 10.1109/LRA.2024.3445635.

[67] S. Wang, Z. Tian, W. Huang, and L. Wang. DDT: Decoupled diffusion transformer. *arXiv preprint arXiv:2504.05741*, 2025.

[68] J. Orbik and F. Ebert. Oculus reader: Robotic teleoperation interface. `https://github.com/rail-berkeley/oculus_reader`, 2021.

[69] K. Zakka. Mink: Python inverse kinematics based on MuJoCo. `https://github.com/kevinzakka/mink`, 2025.

[70] T. Chen, Z. Chen, B. Chen, Z. Cai, Y. Liu, Q. Liang, Z. Li, X. Lin, Y. Ge, Z. Gu, et al. RoboTwin 2.0: A Scalable Data Generator and Benchmark with Strong Domain Randomization for Robust Bimanual Robotic Manipulation. *arXiv preprint arXiv:2506.18088*, 2025.

[71] F. Xiang, Y. Qin, K. Mo, Y. Xia, H. Zhu, F. Liu, M. Liu, H. Jiang, Y. Yuan, H. Wang, L. Yi, A. X. Chang, L. J. Guibas, and H. Su. SAPIEN: A simulated part-based interactive environment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[72] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Robotics: Science and Systems (RSS)*, 2023.

[73] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Robotics: Science and Systems (RSS)*, 2023.

Supplementary Materials

Contents

A Additional Real-World Experiment Results and Analysis

This section provides additional real-world experimental results and analysis. Further visualizations of our real-world rollouts and world-model predictions are available on our website: `gatech-rl2.github.io/egowam.github.io`.

A.1 Ablation on Aligned Human Data

Sec. 5.2(**Q3**) studied one extreme of action alignment: *deliberately misaligned* human data collapses BC below its robot-only baseline, while the 3D-flow WAM stays robust. Here we probe the other extreme: does *manually aligning* the demonstrator to the robot's viewpoint and grasp strategy (Fig. 10(c)) help each paradigm? We co-train on `cup-on-saucer`, the task with the strongest human–robot mismatch (Fig. 1), under two 1:1 human regimes: **In-Domain Co-train** (natural human data, unmatched viewpoint and behavior studied in Sec. 5) and **Aligned Co-train** (human demonstrator intentionally mirrors the robot, described in App. C).

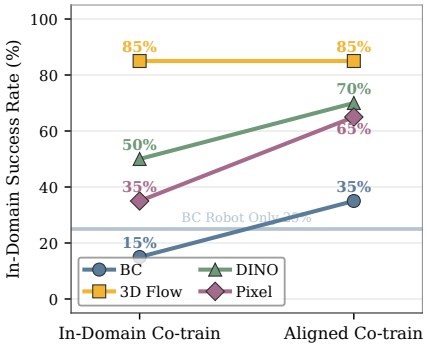


Figure 9: **Ablation of Aligned Human Data.** Aligning the demonstrator lifts BC above robot-only and gains Pixel/DINO 20–30 pts, isolating human head motion; 3D-flow holds at 85% either way.

Fig. 9 reports the difference between In-Domain Co-train and Aligned Co-train across the four variants (BC, Pixel, DINO, 3D Flow). Three findings stand out:

- **Aligned human data lifts BC above its robot-only baseline.** BC drops under natural in-domain human data, reproducing the negative-transfer pattern. When the demonstrator is aligned to the robot, BC effectively gains from human data, confirming that BC can benefit from human data *only* when the action distribution is hand-curated to match the robot's.
- **Pixel and DINO improve when ego-motion is factored out at collection time.** Pixel rises 35% → 65% and DINO 50% → 70% when the human head motion matches the robot's static

(a) Robot Teleoperation (b) Natural Human Demonstration (c) Robot-Aligned Human Demonstration

Figure 10: **Robot Platform and Data Collection. (a)** Our robot platform: two upright-mounted 6-DoF ARX5 arms with two wrist-mounted Intel RealSense D405 cameras and a head-mounted Project Aria headset, teleoperated by a human via a Meta Quest 3 interface. **(b)** A human wearing Project Aria glasses collects demonstrations naturally. **(c)** The human deliberately aligns with the robot's action, height, and viewpoint, producing a static egocentric view matched to the robot platform.

ego-camera. The gap quantifies how much these image-coordinate targets suffer when *D3 (ego-motion factoring)* is violated.

- **The 3D-flow WAM is invariant to alignment.** It holds at 85% under both regimes: the camera-stabilized 3D-flow target factors out ego-motion by construction (Sec. 4.2), so the gains others recover through manual alignment come for free.

A.2 Ablation on Human Data Modality

How much of EGOWAM's gain comes from the *action labels* on human demonstrations versus the *world-model supervision*? We ablate three human co-training modalities on `cup-on-saucer`: **Action Only** (BC co-train baseline), **3D Flow only** (no action supervision on human batches), and **Action + 3D Flow** (full EGOWAM). Fig. 11 yields two findings:

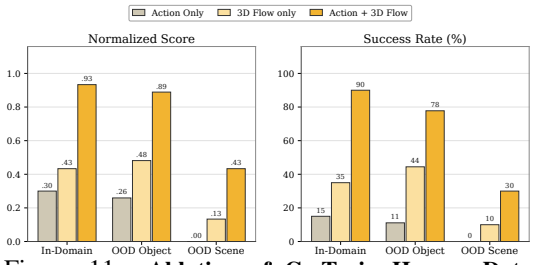


Figure 11: **Ablation of Co-Train Human Data Modality.**

- **World-model supervision alone outperforms action supervision alone.** 3D-flow-only outperforms action-only across all three splits, most starkly on OOD Scene, where action-only fails entirely (0%) while 3D-flow-only retains 10% SR. The trunk shaping induced by predicting 3D motion in human video is a stronger context-level transfer signal than action labels.
- **Action labels and world-model supervision are mutually reinforcing.** Joint training wins on every split, and we attribute this to two complementary effects. (1) *Action as context*: action labels condition the trunk on the demonstrator's intent, sharpening the world model prediction; this effect is measurable directly in the world-model loss, where co-training lowers the 3D-flow prediction loss on both the human and robot streams (Table 1). (2) *Action as task-relevance signal*: action labels mark which motion in the scene is task-relevant, focusing the trunk on agent-caused dynamics rather than incidental flow.

Table 1: 3D-flow world-model prediction loss. Adding action labels (*Action + Flow*, full EGOWAM) lowers the loss over *Flow-Only* on both streams, supporting *action as context*.

Flow stream	Flow-Only	Action + Flow
Human	0.23	**0.22**
Robot	0.20	**0.19**

A.3 Failure Analysis for Pixel-PT on Bag-Grocery

As shown in Fig. 3, Pixel-PT (robot only) underperforms both BC and Pixel on `bag-grocery` task. Here, we provide further failure analysis on it. From our observation, most failures occurred at the

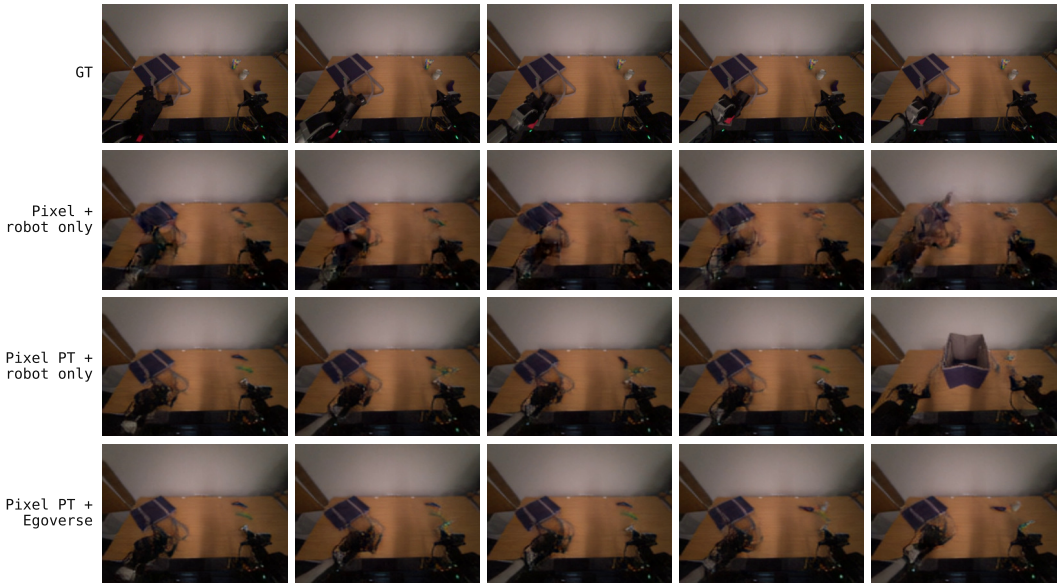


Figure 12: **Video Prediction for Failure Analysis on Bag-Grocery.** 6-step rollouts from the same initial frame. **Pixel + robot only**: blurry but faithful, where the bag stays closed until acted on. **Pixel-PT + robot only**: hallucinates an already-open bag, causing the policy to skip the opening stage. **Pixel-PT + Egoverse**: sharp and faithful, showing human co-training removes the hallucination.

bag-opening stage, and Fig. 12 isolates the cause. **Pixel-PT** renders crisp future frames in which the bag already appears open before the gripper has acted on the handles: the natural-image prior, in which bags are typically open, overrides the actual scene state, and the policy advances to the pick stage prematurely. **Pixel (from-scratch)** produces blurrier frames but tracks bag openness faithfully, and the resulting policy opens the bag more reliably. **Pixel-PT + EgoVerse** keeps the sharp pretrained output while predicting the closed-then-opening progression correctly, indicating that human co-training can correct noisy prior-induced hallucination in pretrained video models by increasing the data amount, and thus improves the policy a lot.

B Additional Implementation Details

B.1 Behavior-Cloning Co-Training Architecture

Stems and Trunk. Each embodiment is tokenized by a shallow, embodiment-specific stem that projects into a shared 256-d space via learned query attention (16 latent queries, 8 heads, head dim 64). The egocentric stream uses a ResNet-18 [63] or pretrained DINO encoder [12, 61] stem shared across embodiments; robot batches add a matching wrist-view stem, and the 14-d end-effector proprioception is encoded by a per-embodiment MLP. A single transformer trunk (256-d, 16 blocks, 8 heads, stochastic depth 0.1, with learned domain embeddings) processes the observation tokens together with 64 learnable action tokens and 16 future tokens, using a one-frame observation history.

Flow-Matching Action Head. The action head is a 6-block CrossTransformer (hidden width 128, 4 heads) trained with the conditional flow-matching objective of Sec. 4.1. It denoises a chunk of 100 action tokens via alternating self- and cross-attention conditioned on the trunk, with the flow-matching timestep drawn from $\mathrm{Beta}(1.5, 1.0)$, and is sampled with 50 v-prediction steps. Cross-embodiment actions are unified to the 14-d end-effector space and quantile-normalized (Sec. 3.1).

B.2 World-Model Heads

All three heads share the linear flow path $s^\tau = (1-\tau)\epsilon + \tau s$, $\tau \in [0,1]$, $\epsilon \sim \mathcal{N}(0, I)$, and differ only in the target s_{t+T} and the head architecture that consumes the trunk embeddings.

Table 2: Hyperparameters held fixed across all world-model variants. Only the world-model head and target (bottom block) change between runs.

Component / Setting	**Value**
Backbone and stems	
Trunk (embed dim / blocks / heads)	$256/16/8$
Query tokens consumed by trunk	64 action + 16 future (WAM); 64 action only (BC)
Ego / wrist vision stem	ResNet-18 [63], output dim 256
Proprioception stem	MLP ($14 \to 256$)
Cross-attn stem (latent / heads / head dim)	$16/8/64$
Observation horizon	1
Stochastic depth (drop path)	0.1
Action head	
Architecture	CrossTransformer (flow matching)
Blocks / hidden dim / heads	$6/128/4$
Action dim d_a	14 (per-arm 6-DoF $\mathrm{SE}(3)$ + gripper)
Resampled chunk length k	100
Robot / human window T_R, T_H	1.5 s (45 frames) / 1.0 s (30 frames)
Prediction type / sampling steps	v-prediction / 50
Flow-matching τ prior	$\mathrm{Beta}(1.5, 1.0)$
Normalization	quantile (1st/99th pct. $\to [-1, 1]$)
Deployment inference rate	30 Hz
Optimization	
Optimizer	AdamW (lr 1×10^{-4}, weight decay 1×10^{-4})
Scheduler	cosine annealing ($T_{\max} = 1400$, $\eta_{\min} = 1\times10^{-5}$)
Batch size (robot / human)	$32/32$
Steps per epoch / max epochs	$100/2000$
Precision	bf16
World-model loss weight λ	1.0
Augmentation (train only)	color jitter $(0.1, 0.1, 0.1, 0.05)$
Image preprocessing	ImageNet normalization
World-model heads (swappable)	
Pixel: VAE / decoder	Wan VAE [59] / DiT, from scratch
decoder (blocks / dim / heads)	$6/384/6$
latent ($C\times H\times W$) / input res	$16\times16\times16$ / 128
Pixel-PT: VAE / decoder	Wan VAE [59] / VACE-1.3B [65], pretrained
decoder (layers / dim / heads / FFN)	$30/1536/12/8960$
latent ($C\times H\times W$) / input res	$16\times16\times16$ / 128
DINO: encoder / pre-processing	DINOv2-B [12], frozen / drop `[CLS]` & registers, per-token LN
denoiser ($\mathrm{DiT}^{\mathrm{DH}}$ [13])	DiT backbone + wide DDT-style head [67]
backbone (blocks / dim / heads)	$6/384/6$
wide head (blocks / dim)	$2/2048$
feature dim ($C\times H\times W$)	$768\times16\times16$
3D Flow: tracker / denoiser	Track4World [15] / flow matching
denoiser (blocks / dim / heads)	$4/256/4$
anchor grid / dim / horizon	28×40 (1120 pts, no subsampling) / 3 / 100
query condition	anchor positions q from current ego frame

VAE Video Prediction Head. The target is the future ego frame (resized to 128×128) encoded in the latent space of a frozen pretrained Wan video VAE [59], $s = \mathrm{VAE}(I^{\mathrm{ego}}_{t+T})$, giving a $16\times16\times16$ latent. We study two instantiations of this head, both trained under flow-matching v-prediction with 50 sampling steps. **Pixel** predicts the latent with a lightweight diffusion transformer (DiT) of 6 blocks, hidden width 384, and 6 heads, patchified at stride 2 ($8\times8 = 64$ tokens) and conditioned on the mean-pooled trunk embedding, trained from scratch. **Pixel-PT** is initialized with the pretrained Wan 1.3B transformer. This backbone has 30 layers, hidden width 1536, 12 attention heads, and FFN dimension 8960. The two variants share the same VAE target and trunk conditioning, and differ in head capacity and initialization.

RAE DINO Prediction Head. The target is the DINOv2-B [12] patch-feature map of the future ego frame, $s = \mathrm{DINO}\big(I^{\mathrm{ego}}_{t+T}\big)$, a 16×16 token grid in $\mathbb{R}^{768}$. We adopt the Representation Autoencoder (RAE) [13] on two fronts: its frozen DINOv2 encoder defines the prediction target, and its $\mathrm{DiT}^{\mathrm{DH}}$ denoiser architecture defines our head. The trained pixel decoder is not used during EGOWAM training – supervision happens directly in feature space and the world-model head is discarded at inference – and is only invoked for visualization. Following RAE, we drop `[CLS]` and register tokens and apply per-token layer normalization before the loss. The $\mathrm{DiT}^{\mathrm{DH}}$ head pairs a standard DiT backbone with a *shallow but wide* DDT-style head [67], motivated by RAE's finding that diffusion in semantic latent spaces requires denoiser width $\geq$ token dimensionality (768 here), below which the flow-matching loss provably fails to converge. We use a 6-block, 384-d backbone (6 heads) and a 2-block, 2048-d wide head, both conditioned on the mean-pooled trunk embedding, and train under v-prediction: $\mathcal{L}^{\mathrm{DINO}}_{\mathrm{world}} = \mathbb{E}\,\|\epsilon - \epsilon_\psi(s^\tau, \tau, f_\phi(o))\|^2$, sampled with 50 fixed Euler steps.

3D Flow Data Processing and Prediction Head. The target is a dense 3D motion field over $[t, t+T]$ expressed in the camera-stabilized frame at time t. We obtain it with Track4World [15], a feed-forward dense 3D point tracker that takes RGB frames alone and predicts per-pixel 3D scene flow together with metric depth, intrinsics, and camera poses. Raw 3D displacement on egocentric video is dominated by ego-motion: stationary objects induce large apparent flow whenever the wearer turns their head. We therefore use the Aria VIO head poses [62] to re-express the predicted future positions X_{t+T} in the camera frame at t, $\tilde{X}_{t+T} = (T^{\mathrm{cam}}_t)^{-1}\, T^{\mathrm{cam}}_{t+T}\, X_{t+T}$, and define the flow target $s = F_{[t,t+T]} = \tilde{X}_{t+T} - X_t$. After this stabilization a static background yields near-zero flow while manipulated objects retain motion proportional to physical displacement; for robot clips the head camera is fixed, so the transform is the identity. The flow is read on a fixed 28×40 (1120-point) pixel-anchor grid, and to suppress tracking noise we discard anchors whose displacement falls below a movement threshold (2 mm for robot tracks; 10 mm for human tracks, which carry residual head motion) and skip the first/last 20 frames of each human clip. The head is a flow-matching decoder (4 blocks, hidden width 256, 4 heads) that, conditioned on the anchor positions q and the trunk features, regresses the 3D displacement of *all* 1120 anchors over the 100-step horizon (target shape $100\times1120\times3$) with no subsampling, $\mathcal{L}^{\mathrm{Flow}}_{\mathrm{world}} = \mathbb{E}\,\|u_\psi(s^\tau, \tau, f_\phi(o), q) - (s_q - \epsilon_q)\|^2$.

B.3 Training and Inference

Training. Each step draws 32 robot and 32 human samples and supervises the shared trunk with $\mathcal{L}_{\mathrm{action}} + \lambda\,\mathcal{L}_{\mathrm{world}}$, $\lambda = 1$ (Eq. 3). We optimize with AdamW (learning rate 1×10^{-4}, weight decay 1×10^{-4}) under cosine annealing ($T_{\max} = 1400$, $\eta_{\min} = 1\times10^{-5}$) in bf16. All variants except Pixel-PT are trained on a **single NVIDIA L40S GPU** for 2000 epochs of 100 steps; Pixel-PT is trained on $2\times$ L40S in data-parallel for 1000 epochs of 100 steps to accommodate the 1.3B-parameter pretrained backbone within memory. Under either configuration, end-to-end training takes approximately two days per task per method. Images are ImageNet-normalized at train and test time, with color jitter (±0.1 brightness/contrast/saturation, ±0.05 hue) added during training only. Remaining settings are listed in Table 2.

Inference. At deployment the world-model head g_ψ is frozen and detached from the computation graph: only the shared trunk f_ϕ and the flow-matching action head π_θ are unrolled to decode $a_{t:t+k}$ from a one-frame observation, consistent with the action-only formulation in Sec. 4.3. The full policy runs on a **single NVIDIA RTX 4090 GPU at** 30 **Hz**, matching the control rate of the underlying ARX5 platform. Discarding the world-model head at inference also keeps the deployment footprint identical across all four world-representation variants, so any rollout differences in Sec. 5 reflect what each target taught the trunk during training rather than test-time compute.

C Additional Real-World Experiment Details

C.1 Robot Platform and Data Collection

As shown in Fig. 10, our bimanual platform comprises two upright-mounted 6-DoF ARX5 arms with parallel-jaw grippers, head-mounted Project Aria Gen-1 glasses [62] providing the egocentric RGB stream shared with human demonstrations, and two wrist-mounted Intel RealSense D405 cameras.

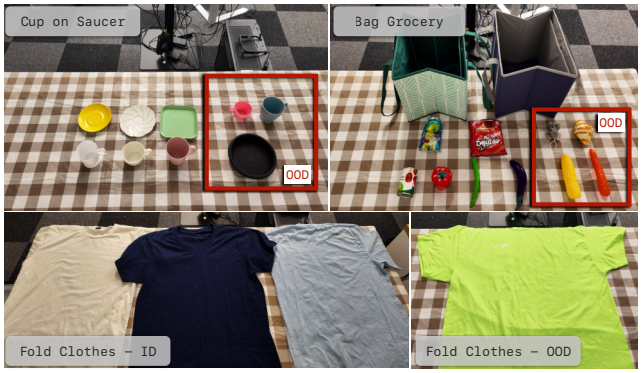


Figure 13: **In-Domain and OOD Objects.**

We collect demonstrations through the RAIL Lab Oculus Reader [68] interface, driving the arms with a Meta Quest 3 headset and Touch Pro controllers. Commanded base-frame end-effector poses are converted to joint angles by the Mink IK solver [69], and the resulting targets are tracked by the ARX5 joint-space controller. The runtime is a multi-threaded Python stack; low-level hardware communication runs over a CAN bus.

Robot actions are per-arm 6-DoF end-effector poses with a 1-D gripper command, $a^R_{t:t+k} \in \mathbb{R}^{k\times 14}$. They are computed from commanded joint angles via forward kinematics, projected into the egocentric Aria camera frame using the calibrated extrinsics, and expressed as $(x, y, z, \text{yaw}, \text{pitch}, \text{roll})$ Euler poses per arm. Following Sec. 3.1, actions are quantile-normalized by mapping each dimension's 1st and 99th percentiles to $[-1, 1]$ to be robust to tracking outliers.

We collect 300–360 teleoperated demonstrations per task, with randomized object placements, orientations, and 4–8 object combinations for each task. Per-task demonstration counts and hours are listed in Table 3, and the training/testing object sets are shown in Fig. 13.

Table 3: Data composition per task. *In-Domain* denotes small-scale, scene- and object-aligned human data; *EgoVerse* denotes the large-scale EgoVerse-A flagship split.

Task	**Robot** (# demos / hours)	**Human (In-Domain)** (hours)	**Human (EgoVerse)** (hours)
cup-on-saucer	300 / 2.5h	2h	20.5h
fold-clothes	360 / 3.0h	2h	21h
bag-grocery	300 / 2.5h	2h	7h

C.2 Human Data Collection and EgoVerse Dataset

In-Domain Human Data Collection Setup. In-domain human data is captured with Project Aria glasses, lightweight (75 g) head-worn devices with a wide-FoV RGB camera and two synchronized monochrome scene cameras used for SLAM. Hand poses (21 keypoints and 6-DoF palm pose per hand) and a calibrated 6-DoF head pose from visual–inertial SLAM are recovered through the Aria Machine Perception Service (MPS) [62]. Following the EgoVerse protocol, demonstrations are recorded in roughly 5-minute units yielding several demonstrations each, within an approximately 40 cm $\times$ 60 cm workspace with object positions randomized across trials. In-domain human data uses the *same* scenes and objects as the robot data but differs in viewpoint and behavior, and is collected at a 1:1 ratio with the robot demonstrations (Table 3).

Aligned and Unaligned Human Data Collection. To probe the robustness of each paradigm to action-(mis)aligned human data (Sec. 5.2(Q3) and App. A.1), we collect two auxiliary human datasets that bracket the alignment axis. Both reuse the in-domain human data collection setup above; only the demonstrator's execution strategy and viewpoint change.

- **Unaligned** (`bag-grocery`)**.** A counterfactual set in which the demonstrator performs the task in a manner whose retargeted trajectory yields *inexecutable* robot actions, e.g., grasps the parallel-jaw gripper cannot reproduce (Fig. 8, right).

- **Aligned (`cup-on-saucer`).** The demonstrator deliberately mimics the robot's motion, with camera height and viewpoint matched to the robot's static Aria mount (Fig. 10). This isolates execution alignment from sensing alignment by collapsing the latter.

EgoVerse Dataset. Our large-scale, in-the-wild human regime is the EgoVerse-A flagship split [7], which spans diverse scenes, objects, and demonstrators with *no* scene or object alignment to the robot data. We use the per-task flagship subsets for `cup-on-saucer`, `fold-clothes`, and `bag-grocery`, giving an overall $\sim$10:1 human-to-robot ratio (Table 3). Each demonstration carries 3D hand pose for both hands (21 keypoints per hand in the camera frame) paired with a calibrated 6-DoF head pose from visual–inertial SLAM, which we use both to align the action channel and to stabilize the 3D-flow target into the camera frame at t (Sec. 4.2).

C.3 Rollout Evaluation Protocol

Experiment Methods. Each method is evaluated on **ID** (20 rollouts: seen objects and scene, randomized positions and orientations) and **OOD** (20 rollouts: 10 unseen objects in the training scene, 10 seen objects in novel scenes with varied backgrounds and table heights), for **1800** rollouts in total across all methods and tasks. For every task we define task-specific sub-task metrics (grasps, placements, intermediate manipulations, and full completion) and report a normalized sub-task score aggregated across rollouts alongside the binary success rate. Initial conditions are randomized and held common across methods within each task to ensure a fair comparison.

Cup-on-Saucer. The robot must reorient a cup from a randomized initial pose and place it upright on a saucer at a randomized position, demanding precise bimanual regrasping and fine-grained transport. Sub-task credit (1 point each, 3 total) is assigned for (i) rotating and picking up the cup, (ii) a successful handover between the two arms, and (iii) placing the cup upright on the saucer. SR is the fraction of rollouts in which the cup is correctly placed on the saucer.

Fold-Clothes. The robot must three-fold a T-shirt initialized in random configurations, a deformable task with shape variation and self-occlusion across stages. Sub-task credit (1 point each, 3 total) is assigned for (i) the bottom-sleeves fold, (ii) the top-sleeves fold, and (iii) the final fold in half. SR requires all three stages to complete cleanly.

Bag-Grocery. The robot must open a grocery bag and load three items into it from randomized positions—a long-horizon task in which the two arms first grasp the handles to open the bag, then insert items one by one. Sub-task credit is 1 point for opening the bag and 1 point per item placed inside, for a maximum of 4 points per rollout; an insertion counts only if the objects are placed in left-to-right order inside the bag. SR requires all three items loaded under the ordering constraint. Because its pick-and-place motions are naturally aligned between human and robot, this is the one task where action-level co-training helps, which we exploit in the Sec. 5.2(Q3) ablation.

D Robot-to-Robot Transfer in Simulation: RoboTwin

Our real-world experiments (Sec. 5) establish EgoWAM's two central claims: that cross-embodiment co-training transfers across the human–robot gap, and that the world-model head is the interface that drives this transfer. To validate the same claims in a *public, reproducible, robot-to-robot* setting, we instantiate EgoWAM on the RoboTwin 2.0 bimanual benchmark [70], which ships teleoperated demonstrations for multiple robots on the same tasks. The goal here is not a new benchmark result but a controlled replication: under a shared, dimension-invariant end-effector action space, does cross-embodiment co-training still beat single-embodiment training, and does the world-model head still help to transfer with abstraction?

D.1 Simulation Setup

RoboTwin 2.0 runs on SAPIEN [71]. We use the bimanual `aloha-agilex` (two 6-DoF arms) as the primary embodiment and `arx-x5`, `franka`, `ur5` as co-training embodiments, where `franka` and

ur5 are two 7-DoF arms, on three hard tasks: pick-diverse-bottles (15-instance bottle pool), stack-bowls-three (long-horizon manipulation), and hanging-mug (fine-grained manipulation). Each uses the shipped demo_clean-50 split.

D.2 Integration and Baselines

We add each robot to the tokenizer stack, reusing the trunk and action head. Because the arms differ in DoF, we predict for both arms an absolute end-effector action in a head-camera frame—$[\mathbf{p}_{xyz}, \boldsymbol{\theta}_{\text{ZYX}}, g]$ per arm (14-D)—identical across robots, so a single action head serves every embodiment and cross-embodiment co-training is well-posed, exactly as in our real-world formulation (Sec. 3.1). At test time the predicted chunk is resampled to the control rate and streamed to a differential-IK controller [69] under a receding horizon. Per task we train every world-model variant *single* (aloha-agilex only) and *cross* (all four robots), holding trunk, head, and optimizer fixed. As same-action-space references we adapt RoboTwin's official ACT [72] and Diffusion Policy [73] pipelines to our setting (**ACT-EE** and **DP-EE**) by replacing their native joint-space state and action with the same 14-D EE vector and evaluating through the identical IK executor; they are single-embodiment and world-model-free, so they isolate what cross-embodiment co-training and the world head add on top of the action space alone. We evaluate all methods on 100 held-out seeds ($\geq 10^5$, sparse success, identical across methods).

D.3 Results and Analysis

Table 4: RoboTwin held-out closed-loop success (%, 100 seeds). **Left:** same-action-space, single-embodiment references (ACT-EE, DP-EE). **Right:** EGOWAM variants, each trained single (*s*) and cross-embodiment (*c*), sharing the trunk and end-effector action head and differing only in the world target. Best per task in **bold**. †stack is evaluated on an appearance-shifted object (see text).

	Baselines		**BC**		**Pixel**		**DINO**		**3D Flow**	
Task	ACT-EE	DP-EE	s	c	s	c	s	c	s	c
pick-diverse-bottles	2	5	2	6	7	11	4	**28**	0	16
stack-bowls-three†	0	0	0	0	0	0	0	8	0	**16**
hanging-mug	0	0	0	0	0	**1**	0	0	0	0

Table 4 reproduces the main paper's findings in robot-to-robot simulation. Four points stand out:

- **Cross-embodiment co-training beats single-embodiment training.** Every EGOWAM variant improves from single to cross (DINO $4 \to 28$, Pixel $7 \to 11$, BC $2 \to 6$ on pick), and cross is the *only* setting that succeeds at all on stack. Since single-embodiment policies already overfit their training scenes yet barely generalize, the gap is genuine cross-embodiment transfer rather than better fitting, the same mechanism as our real-world results (Sec. 5).
- **The world-model head, not the action space, drives the transfer.** Under cross-embodiment training the world-model variants (Pixel/DINO/3D Flow) consistently exceed the action-only BC policy. The same-action-space but world-model-free references ACT-EE/DP-EE stay at or below the single-embodiment variants ($\leq 5\%$ on pick, 0% elsewhere); since they change only the policy architecture, the gain is attributable to co-training through the world-model interface.
- **DINO and 3D Flow are strong at object generalization on the first two tasks.** Both pick and stack stress object generalization, and the appearance-abstracting targets lead. On pick (a 15-instance bottle pool) DINO reaches the best 28% and 3D Flow 16%. On stack, an incidental asset change between the released demonstrations and the current simulator (the bowl's material shifted while its pose and geometry did not, and demo_clean adds no texture randomization) turns evaluation into an appearance-shift test; *only* the appearance-invariant DINO (8%) and 3D Flow (16%) survive it while BC and Pixel collapse to 0%, directly reflecting the appearance-abstraction criterion (D1) of Sec. 4.2.
- **Very fine-grained manipulation still fails for all methods.** hanging-mug requires threading a mug onto a thin rack (a millimeter-precise insertion) and here *every* method, including DINO

and 3D Flow, stays at $\leq 1\%$. The bottleneck on this task is manipulation precision, not object generalization or the world target: the appearance abstraction that carries `pick` and `stack` does not by itself supply the sub-centimeter accuracy this insertion demands. Closing this gap is orthogonal to the transfer gains above and is left to future work.