

ReWeight: Leveraging Human Data for VLA

Post-Training via Demonstration Retrieval and Sample Weighting

Chenwei Wang, Dianye Huang, Match W.L. Ko, Chenjia Bai, and Zhongliang Jiang

Abstract—Post-training vision-language-action (VLA) models for specific robots and tasks requires in-domain demonstrations, yet collecting diverse robot data is costly. Egocentric human demonstrations provide a scalable alternative, but directly mixing human and robot data can introduce cross-embodiment discrepancies and degrade policy performance. To address this challenge, we introduce ReWeight, a framework that incorporates human data into VLA post-training through demonstration-level retrieval and sample-level weighting. ReWeight learns a cross-embodiment visuomotor representation that combines visual observations with future actions to measure behavioral similarity between human and robot demonstrations. Based on optimal transport, it retrieves human demonstrations relevant to the target robot data and assigns larger weights to samples with smaller cross-embodiment discrepancies. We evaluate ReWeight using $\pi_{0.5}$ across eight simulation tasks and four real-world tasks under both clean and randomized settings. In simulation, ReWeight improves the average success rate of post-trained $\pi_{0.5}$ from 39% with only robot data and 44% with randomly mixed human-robot data to 57%. In the physical experimental setting, it achieves an average success rate of 68.8%, outperforming the baselines by 28.8% and 13.8%, respectively. Overall, ReWeight provides an effective paradigm for transforming abundant egocentric human experience into transferable supervision for robot learning. (Project webpage: <https://reweight-vla.github.io>)

Index Terms—Vision-Language-Action Models, Egocentric Human Demonstrations, Data Retrieval, Robot Manipulation

I. INTRODUCTION

VISION-language-action (VLA) models have shown strong promise for general-purpose robotic manipulation by integrating visual perception, language understanding, and action generation within a unified policy [1], [2], compared to conventional control approaches [3]–[5]. Scaling VLAs with diverse robot data has enabled increasingly broader generalization [1], [6]. However, deploying a pretrained VLA on a specific task or robot platform often relies on post-training with in-domain data to ground the policy in the target task, environment, and embodiment [2], [6]. Therefore, the performance of a VLA remains closely tied to the availability and diversity of embodied data for adaptation.

Collecting such data directly on robots requires task-specific hardware setup and repeated interaction with the environment,

(Corresponding author: Zhongliang Jiang)

Chenwei Wang, Dianye Huang, Match W.L. Ko, and Zhongliang Jiang are with the Department of mechanical Engineering, The University of Hong Kong, Hong Kong SAR, China. (e-mail: chenweiwang@connect.hku.hk, dianye.huang@hku.hk, matchko@hku.hk, zljiaang@hku.hk).

Chenjia Bai is with the Institute of Artificial Intelligence (TeleAI), China Telecom, Shanghai 200232, China, and also with the Shenzhen Research Institute of Northwestern Polytechnical University, Xi'an 710072, China (e-mail: baicj@chinatelecom.cn).

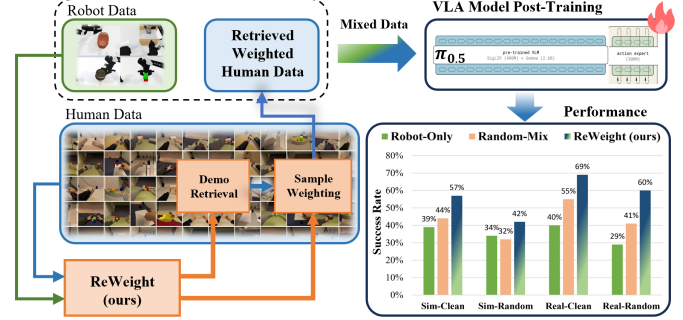


Fig. 1: Illustration of ReWeight, retrieving human demonstrations and assigning sample weights for VLA post-training.

making it costly and difficult to acquire data with the scale and diversity demanded by the large models. In contrast, humans, representing the most prevalent embodiment in the physical world, can continuously generate diverse manipulation data at a scale far beyond what is feasible through robot data collection [7], [8]. Human data has therefore attracted growing interest as a scalable and cost-effective source of embodied supervision for robot learning. In particular, EgoVLA and Being-H0 demonstrated that large-scale human videos can provide useful priors for subsequent adaptation to robotic platforms [9], [10]. Besides, EgoMimic showed that jointly training on aligned human and robot demonstrations can substantially improve imitation learning [11]. More recently, Kareer *et al.* systematically showed that sufficiently large and diverse VLA pretraining can yield increasingly embodiment-agnostic representations, enabling human-to-robot transfer even without explicit cross-embodiment correspondence engineering [12]. Being-H0.5 further extended it toward large-scale cross-embodiment learning across human and robotic embodiments [13].

However, simply scaling heterogeneous data does not guarantee better robot policies. Wang *et al.* showed that indiscriminately mixing data across embodiments can introduce interference and even degrade VLA performance [14]. Similarly, Punamiya *et al.* found that the benefit of scaling human data depends critically on its alignment with the robot learning objective [15]. These findings highlight the limitations of indiscriminate data scaling and motivate selective use of human data. Retrieval provides a direct mechanism for such selection. Prior approaches have exploited task similarity to retrieve relevant robot data from large offline datasets, including the use of lightweight human hand demonstrations as queries

for task-relevant robot play data [16]. Retrieval from large egocentric video banks has also been explored to provide task-relevant human data for robotic manipulation [17]. SiMDex formulates human-data selection for VLA post-training as a recommendation problem and mines task-relevant egocentric samples from a large human-data pool, demonstrating advantages over randomly mixing human data [18].

Yet retrieval alone cannot fully capture the varying relevance of human demonstrations to a target robot task. Even within the retrieved subset, demonstrations can differ substantially in their transferability to the target robot. Treating all retrieved samples equally overlooks this fine-grained variation and allows marginally relevant demonstrations to influence optimization as strongly as highly aligned ones. Sample weighting has been explored in robot learning to handle heterogeneous supervision, for example by emphasizing informative interventions or reweighting logged robot transitions during offline post-training [19], [20]. These methods, however, primarily estimate sample importance within the robot domain based on factors such as data quality or informativeness. In human-to-robot transfer, sample importance additionally depends on the cross-embodiment relevance of each human demonstration to the target robot behavior. Effective use of large-scale human data therefore requires jointly addressing two coupled decisions: (i). *which human demonstrations should be transferred?*, and (ii). *how much should each selected sample contribute to VLA post-training?*

To this end, we introduce ReWeight, a unified framework for leveraging human data for VLA post-training through demonstration-level retrieval and sample-level weighting. ReWeight first learns a cross-embodiment visuomotor representation that enables human and robot demonstrations to be compared despite differences in appearance and embodiment. Given a small set of robot data and a large pool of human data, ReWeight retrieves an equal size of human data that is most relevant to the target robot data. Crucially, rather than assigning all retrieved samples equal importance, ReWeight further estimates their relative cross-embodiment relevance and converts it into continuous sample-level weights during VLA post-training. Highly aligned human data therefore provides stronger supervision, while the influence of less transferable samples is suppressed. In this way, ReWeight moves beyond post-training with only robot data or naive mixture of human-robot data, turning abundant human data into selectively retrieved and sample-specific weighted supervision.

The contributions of this work are summarized below:

- We formulate the use of human data for VLA post-training as a joint retrieval-and-weighting problem, distinguishing between which human demonstrations should be transferred and how strongly each selected sample should influence learning.
- We introduce ReWeight, a framework that learns a unified visuomotor representation to retrieve demonstration-level human data and derive sample-level weights according to their relevance to target robot data, enabling selective use of heterogeneous human data during VLA post-training.

To evaluate the superiority of ReWeight, we conducted extensive experiments across 8 simulation tasks in the RoboTwin 2.0 benchmark and 4 real-world tasks on the DoBot dual arm platform under the clean and randomized settings. Compared to post-training with only robot data, ReWeight improves the average success rate from 39% to 57% in simulation and from 40.0% to 68.8% in the real world. These results demonstrate the importance of selective retrieval and fine-grained relevance weighting for effectively exploiting egocentric human data in VLA post-training.

II. PRELIMINARY

A. Problem Description

Given M robotic demonstrations $\mathcal{D}_r$ and $N(> M)$ human videos $\mathcal{D}_h^{\text{raw}}$, our goal is to select a subset of the human data $\mathcal{S}_h^*$ that is most similar to the target robot data. Such that, post-training the VLA model on the combined mixture data, $\mathcal{D}_{\text{mix}} = \mathcal{S}_h^* \cup \mathcal{D}_r$, can enable the resulting policy to maximally leverage relevant human manipulation diversity while boosting downstream task performance. This VLA post-training method is formulated as below:

$$\mathcal{S}_h^* = \arg \max_{\mathcal{S}_h \subseteq \mathcal{D}_h, |\mathcal{S}_h|=M, N>M} \text{Sim}(\mathcal{S}_h, \mathcal{D}_r), \quad (1)$$

where the datasets are defined as:

$$\mathcal{D}_r = \left\{ \mathbf{d}^{r(j)} = (\mathbf{o}^r, \mathbf{a}^r, \ell^r)^{(j)} \right\}_{j=1}^M,$$

$$\mathcal{D}_h^{\text{raw}} = \left\{ \mathbf{d}^{h(i)} = (\mathbf{o}^h, \tilde{\mathbf{a}}^h, \ell^h)^{(i)} \right\}_{i=1}^N,$$

where $\mathbf{d}^{r/h}$ denotes the robot/human demonstration, and the superscript $(\cdot)$ represents the index of demonstrations. $\mathbf{o}^{h/r}$ represents the human/robot visual observations, $\mathbf{a}^r$ denotes the control commands for the robot, $\tilde{\mathbf{a}}^h$ denotes the estimated hand kinematics from human videos, and $\ell^{h/r}$ is the corresponding textual command of the human/robot data. $\mathcal{S}_h^* = \{(\mathbf{o}_h, \mathbf{a}_h, \ell_h, \mathbf{w}_h)^{(i)}\}_{i=1}^N$ represents an optimal subset extracted from $\mathcal{D}_h^{\text{raw}}$, where each sample is assigned a weight. This subset is constrained to a fixed size M that maximizes a semantic similarity metric $\text{Sim}(\cdot, \cdot)$.

B. Optimal Transport for Data Discrepancy Measurement

In this work, to measure the similarity, we use optimal transport (OT) to quantify both the *demonstration-level* discrepancy between human and robot demonstrations, and the *sample-level* discrepancy between the human sample and robot demonstrations. These measures will be used to guide human demonstration retrieval and sample weighting during training.

Given a pair of human and robot demonstrations $(\mathbf{d}^h, \mathbf{d}^r)$, we quantify their discrepancy using the entropically regularized *Wasserstein distance*, $\mathcal{W}(\cdot, \cdot)$, defined as:

$$\mathcal{W}(\mathbf{d}^h, \mathbf{d}^r, \mathbf{P}) = \sum_{i=1}^n \sum_{j=1}^m C_{i,j} \mathbf{P}_{i,j} - \epsilon \mathcal{H}(\mathbf{P}), \quad (2)$$

where $\mathbf{P} \in \mathbb{R}^{n \times m}$ is the transport plan, and n and m are the number of timesteps in the human and robot demonstrations,

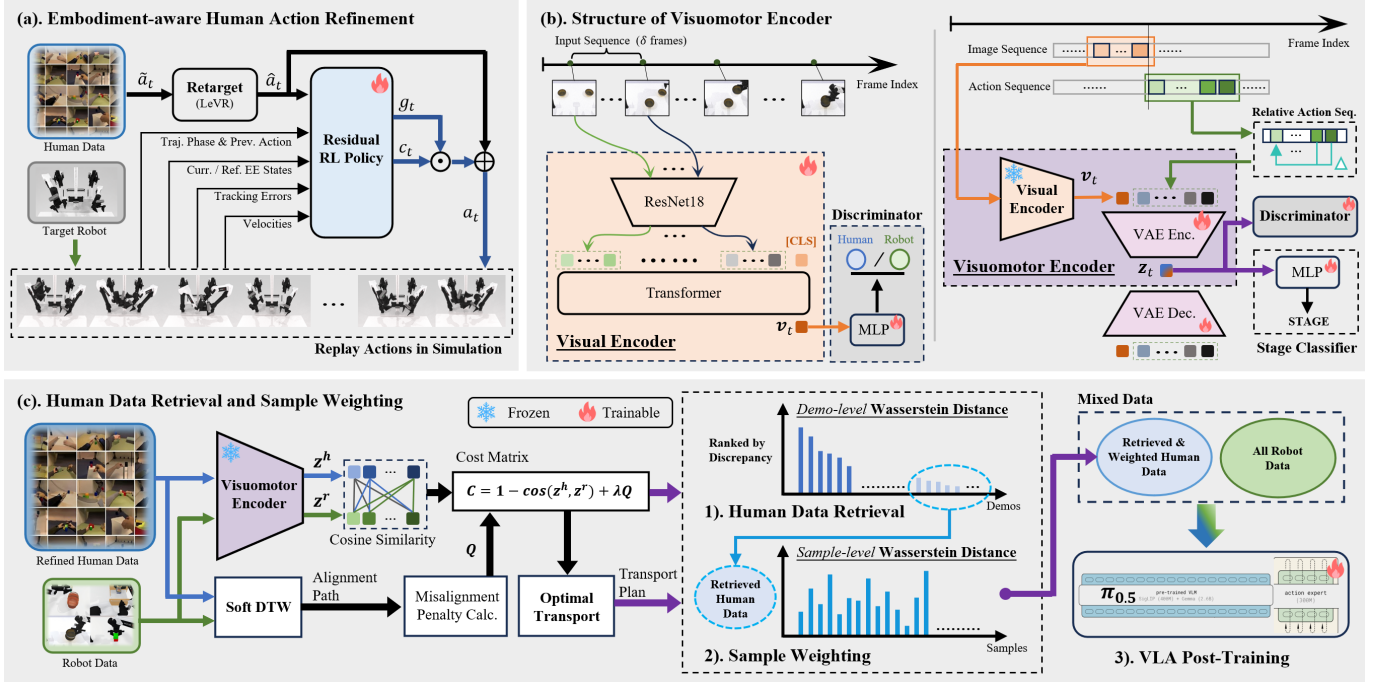


Fig. 2: Overview of the proposed ReWeight framework.

respectively. The cost matrix $\mathbf{C} \in \mathbb{R}^{n \times m}$ specifies the pairwise transport costs, where $C_{i,j}$ represents the cost of matching the human observation at the i -th timestep with the robot observation at the j -th timestep. $\mathcal{H}(\mathbf{P})$ denotes the entropy of the transport plan, and ϵ controls the strength of the entropy regularization. The optimal transport plan is obtained by solving Eq. (3) using the *Sinkhorn* algorithm [21].

$$\begin{aligned} \mathbf{P}^* &= \arg \min_{\mathbf{P} \in \mathcal{M}} \mathcal{W}(\mathbf{d}^h, \mathbf{d}^r, \mathbf{P}), \\ \mathcal{M} &= \{\mathbf{P} : \mathbf{P}\mathbf{1}_n = \mathbf{1}_n/n, \mathbf{P}^T\mathbf{1}_m = \mathbf{1}_m/m\}, \end{aligned} \quad (3)$$

where $\mathcal{M}$ is the set of admissible transport plans with uniform marginal distributions.

Given the optimal transport plan, the *demonstration-level* discrepancy $\bar{\mathcal{W}}$ between the human and robot demonstrations and the *sample-level* discrepancy $\bar{\mathcal{W}}_i$ associated with the i -th human timestep are quantified, respectively, as below:

$$\bar{\mathcal{W}}(\mathbf{d}^h, \mathbf{d}^r) = \sum_{i=1}^n \mathcal{W}_i, \quad \bar{\mathcal{W}}_i(\mathbf{d}_i^h, \mathbf{d}^r) = \sum_{j=1}^m C_{i,j} \mathbf{P}_{i,j}^*. \quad (4)$$

III. METHOD

This section details ReWeight, a framework for leveraging egocentric human data for VLA post-training. As illustrated in Fig. 2, to bridge the human-robot embodiment gap, we refine the extracted hand kinematics and learn a cross-embodiment visuomotor representation for measuring human-robot similarity. The resulting similarity scores are used to retrieve relevant human demonstrations and derive sample-specific weights for VLA post-training.

A. Embodiment-Aware Human Action Refinement

As shown in Fig. 2 (a), to coarsely align human motion with the robot embodiment, we employ LeVR [22] to map the estimated human hand kinematics to the robotic action space, yielding retargeted trajectory $\hat{\mathbf{a}}^h$, where the gripper state is inferred directly from the distance between the human thumb and fingertips. To further resolve the violation of the dynamic constraints such as velocity limits, acceleration bounds, etc., we refine $\hat{\mathbf{a}}^h$ with reinforcement learning. We train a residual policy on top of the retargeted trajectory [23] sampled via the reference state initialization (RSI) [24] technique. At each timestep, the policy predicts both a residual correction c_t and a non-negative gate g_t to control the magnitude of the correction.

$$\mathbf{a}_t = \hat{\mathbf{a}}_t + g_t \cdot c_t, \quad g_t = \text{sigmoid}(s_t) \in (0, 1), \quad (5)$$

where s_t denotes the input state including the current and reference end-effector states, tracking errors, velocities, trajectory phase, and previous action. Following the reward design adopted in BeyondMimic [25], the per-step reward, combining end-effector pose tracking, velocity matching, and regularization terms, is formulated as:

$$r_t = r_{\text{pose}} + r_{\text{vel}} + r_{\text{reg}}.$$

Ultimately, this process yields a refined human dataset with robot-embodiment-aware motion $\mathcal{D}_h = \{(\mathbf{o}_h, \mathbf{a}_h, \ell_h)^{(i)}\}_{i=1}^N$.

B. Cross-Embodiment Visuomotor Encoder

To obtain a unified representation of the robot and human data for subsequent retrieval and sample weighting, we develop a visuomotor encoder comprising a visual encoder and a visuomotor fusion module, as shown in Fig. 2 (b).

1) *Visual Encoder Pretraining*: The visual encoder $\mathbf{E}(\cdot)$ comprises a ResNet-18 backbone and a temporal Transformer. It encodes a sequence of input images $\mathbf{o}_{t-\delta:t}$ into a temporal visual representation $\mathbf{v}_t = \mathbf{E}(\mathbf{o}_{t-\delta:t})$, where $\delta = 8$ represents the length of the input image sequence. The encoder is optimized in two stages: it is first warmed up with Eq. (6) to capture task-relevant visual features, then aligned across human and robot demonstrations via optimizing Eq. (7) to obtain a domain-agnostic representation.

a. *warm-up stage*: To capture task-relevant visual features, the visual encoder is trained from scratch using the following warm-up objective:

$$\mathcal{L}_{\text{warmup}} = \lambda_{\text{sw}} \mathcal{L}_{\text{SwAV}} + \lambda_{\text{tcn}} \mathcal{L}_{\text{tcn}} + \lambda_{\text{tcc}} \mathcal{L}_{\text{tcc}} \quad (6)$$

where $\mathcal{L}_{\text{SwAV}}$ [26] enforces spatial consistency by contrasting cluster assignments of different augmented views of the same frame; $\mathcal{L}_{\text{tcn}}$ [27] denotes the time-contrastive loss. To align coarse task-relevant dynamics across varying sequence lengths, a temporal cycle-consistency loss $\mathcal{L}_{\text{tcc}}$ [28] is applied.

b. *alignment stage*: To align visual representations across human and robot embodiments, we extend the warm-up objective with domain-adversarial learning to suppress embodiment-specific visual cues and promote domain-invariant representations, together with an OT-based correspondence loss, which establishes soft cross-embodiment correspondences and brings task-consistent observations closer in the latent space. The overall visual alignment objective is defined as:

$$\mathcal{L}_{\text{align}} = \lambda_{\text{tcn}} \mathcal{L}_{\text{tcn}} + \lambda_{\text{tcc}} \mathcal{L}_{\text{tcc}} + \lambda_{\text{ot}} \mathcal{L}_{\text{ot}} + \lambda_{\text{adv}} \mathcal{L}_{\text{adv}} \quad (7)$$

where $\mathcal{L}_{\text{adv}}$ denotes the adversarial loss. $\mathcal{L}_{\text{ot}}$ represent the OT correspondence loss, defined as the expected discrepancy over sampled human-robot demonstration pairs:

$$\mathcal{L}_{\text{ot}} = \mathbb{E}_{\mathbf{o}_{t:t-\delta}^h \sim \mathcal{D}^h} [\bar{\mathcal{W}}(\mathbf{v}_t^h, \mathbf{v}^r)], \quad \mathbf{v}_t^{h/r} = \mathbf{E}(\mathbf{o}_{t-\delta:t}^{h/r}). \quad (8)$$

To compute $\mathcal{L}_{\text{ot}}$, we first construct a transport cost matrix $\mathbf{C}$ [see Eq. (9)]. With this matrix, we then solve the entropically regularized OT problem in Eq. (3) to obtain the optimal transport plan $\mathbf{P}^*$ and compute the corresponding OT discrepancy $\bar{\mathcal{W}}$ using Eq. (4). Note that, in our formulation, $\mathbf{C}$ combines visual feature similarity with a temporal mismatch penalty $\mathbf{Q}$:

$$\mathbf{C}_{i,j} = 1 - \cos(\mathbf{v}_i^h, \mathbf{v}_j^r) + \lambda_{\text{dtw}} \mathbf{Q}_{i,j}, \quad \mathbf{Q}_{i,j} = \frac{|j - p_i|}{(m-1)}, \quad (9)$$

where $\mathbf{v}_i^h$ and $\mathbf{v}_j^r$ denote the visual features of the i -th human frame and the j -th robot frame, respectively. The index p_i denotes the robot frame aligned with the i -th human frame according to the alignment path estimated by soft-DTW [29]. Accordingly, $\mathbf{Q}_{i,j}$ penalizes transport between observations that are far apart from the estimated temporal correspondence.

2) *Visuomotor Representation Learning*: To account for motion compatibility between human and robot demonstrations, we construct a visuomotor encoder that incorporates motion information into the visual representation. The encoder follows an InfoVAE-based architecture [30], where the visual representation and its corresponding future action chunk are

encoded by separate MLPs and subsequently fused into a latent visuomotor representation $\mathbf{z}$. The future action chunk $\mathbf{a}_{t:t+L}$, with L empirically set to 20, is represented relative to its initial pose while preserving the original gripper state.

A stage prediction head, $\hat{y} = g_{\text{stage}}(\mathbf{z})$, is used to preserve task-stage information, while an auxiliary loss $\mathcal{L}_{\text{aux}}$ promotes cross-embodiment alignment within the same stage and separation across different stages. The latent objective is:

$$\mathcal{L}_{\text{lat}} = \mathcal{L}_{\text{ce}} + \lambda_{\text{aux}} \mathcal{L}_{\text{aux}} \quad (10)$$

where $\mathcal{L}_{\text{ce}} = \text{CE}(y, \hat{y})$ denotes the cross entropy loss, and

$$\mathcal{L}_{\text{aux}} = \mathbb{E} [-\cos(\mathbf{z}_s^h, \mathbf{z}_s^r)] + \mathbb{E} [\max(0, \cos(\mathbf{z}_s, \mathbf{z}_{s'}) - \mu)],$$

where $\mathbf{z}_s^h$ and $\mathbf{z}_s^r$ denote latent representations from human and robot demonstrations at the same stage s , respectively, while $\mathbf{z}_s$ and $\mathbf{z}_{s'}$ denote representations from different stages, with $s \neq s'$. The first term maximizes the similarity between same-stage cross-domain representations, whereas the second penalizes the similarity between different-stage representations when it exceeds the margin $\mu = 0.3$. The overall VAE training objective is formulated as:

$$\mathcal{L}_{\text{vae}} = \lambda_{\text{rec}} \mathcal{L}_{\text{rec}} + \lambda_{\text{reg}} \mathcal{L}_{\text{reg}} + \lambda_{\text{lat}} \mathcal{L}_{\text{lat}}, \quad (11)$$

where $\mathcal{L}_{\text{rec}}$ is the reconstruction loss for the visual feature and relative motion trajectory. The regularization term $\mathcal{L}_{\text{reg}}$ follows InfoVAE [30] combining the KL divergence and maximum mean discrepancy. The latent objective $\mathcal{L}_{\text{lat}}$ further enforces the latent space according to demonstration stage while reducing stage-level discrepancies between human and robot data. These objectives encourage the latent representation to retain the visual and motion information required for reconstruction, suppress embodiment-specific variations, and preserve the temporal progression of the manipulation task.

C. Human Demonstration Retrieval and Sample Weighting

We employ a two-level strategy for *demonstration-level* retrieval and *sample-level* weighting of the human data for VLA post-training.

1) *Demonstration-level Retrieval*: For each candidate human demonstration $\mathbf{d}^{h(i)}$, we compute its discrepancy from every robot demonstration of the same task. Let

$$\mathbf{s}^{(i)} = [s_1^{(i)}, \dots, s_M^{(i)}] \in \mathbb{R}^M, \quad s_j^{(i)} = \bar{\mathcal{W}}(\mathbf{d}^{h(i)}, \mathbf{d}^{r(j)}),$$

where M is the number of robot demonstrations in each task, and $\bar{\mathcal{W}}(\cdot, \cdot)$ is the demonstration-level discrepancy defined in Eq. (4). When constructing the cost matrix in Eq. (9), we compute the cosine distance using the latent features $\mathbf{z}$ produced by the visuomotor encoder developed in *Sec. III-B*. We sort the elements of $\mathbf{s}^{(i)}$ in ascending order as $s_{(1)}^{(i)} \leq \dots \leq s_{(M)}^{(i)}$ and define the retrieval score of $\mathbf{d}^{h(i)}$ as the average of its K smallest discrepancies: $\bar{s}^{(i)} = \sum_{k=1}^K s_{(k)}^{(i)} / K$. We empirically set $K = 5$ in all experiments as in [31]. After ranking the human demonstrations by $\bar{s}^{(i)}$, we select the M lowest-scoring ones to construct $\mathcal{S}_h^*$, yielding a 1:1 human-to-robot data ratio for VLA post-training as in [12].

2) *Sample-level Weighting*: After retrieval, individual frames within the retained demonstrations may still differ in their relevance to the robot data. We thereby assign a discrepancy-aware weight to each retained human sample. For a human sample at timestep t in $\mathbf{d}^{h(i)} \in \mathcal{S}_h^*$, its sample-level discrepancy is defined as:

$$d_t^{(i)} = \sum_{j \in \mathcal{N}_K^{(i)}} \mathcal{W}_t(\mathbf{d}_t^{h(i)}, \mathbf{d}^{r(j)}) / K, \quad (12)$$

where $\mathcal{N}_K^{(i)}$ denote the indices of its K nearest robot demonstrations identified during retrieval. $\mathcal{W}_t(\cdot, \cdot)$ denotes the frame-level discrepancy defined in Eq. (4). A smaller value of $d_t^{(i)}$ indicates that the human sample is more consistent with the corresponding robot demonstrations.

To assign larger weights to samples with smaller cross-embodiment discrepancies, we first convert the discrepancy into an unnormalized similarity weight:

$$\tilde{w}_t^{(i)} = d_{\max} - d_t^{(i)}, \quad d_{\max} = \max_{(u, l) \in \mathcal{I}_h^*} d_l^{(u)}, \quad (13)$$

where $\mathcal{I}_h^*$ means the set of all samples contained in the retrieved human demonstrations. The normalized weight is then computed as:

$$w_t^{(i)} = \alpha + (1 - \alpha) \cdot \tilde{w}_t^{(i)} / \max_{(u, l) \in \mathcal{I}_h^*} \tilde{w}_l^{(u)}, \quad (14)$$

where $\alpha \in [0, 1]$ is a sample-weight calibration parameter, that ensures sufficient influence of human samples.

3) *Loss for Post-Training*: During post-training, the model is optimized using the mixed robotic data $\mathcal{D}_r$ and the retrieved human data from $\mathcal{S}_h^*$. Let $\ell_i(\theta)$ denote the per-sample VLA loss for sample i . The post-training loss is defined as:

$$\mathcal{L}_{\text{VLA}} = \left(\sum_{i \in \mathcal{D}_r} \ell_i(\theta) + \sum_{i \in \mathcal{S}_h^*} w_j \ell_j(\theta) \right) / |\mathcal{D}_{\text{mix}}| \quad (15)$$

where $\mathcal{D}_{\text{mix}} = \mathcal{D}_r \cup \mathcal{S}_h^*$. $|\mathcal{D}_{\text{mix}}|$ denotes the number of samples. The human sample weight w_j is computed by Eq. (14).

D. Implementation Details

1) *Residual RL Policy*: The residual RL policy for human action refinement adopts an actor-critic architecture and is optimized with PPO [32]. We use a learning rate of 1×10^{-4} , a batch size of 64, and a discount factor of 0.95. For each task, we train a separate residual policy for 50k environment steps, using RoboTwin 2.0 for the AgileX platform and MuJoCo for the DoBot platform. After training, the policy is frozen and applied to refine all retargeted human trajectories before visuomotor representation learning.

2) *Visuomotor Encoder*: We randomly sample human demonstrations to match the number of available robot demonstrations for each task, and combine the sampled human data with all robot demonstrations for training. The visuomotor encoder is optimized with AdamW using a learning rate of 1×10^{-4} and a batch size of 64. Training proceeds in three

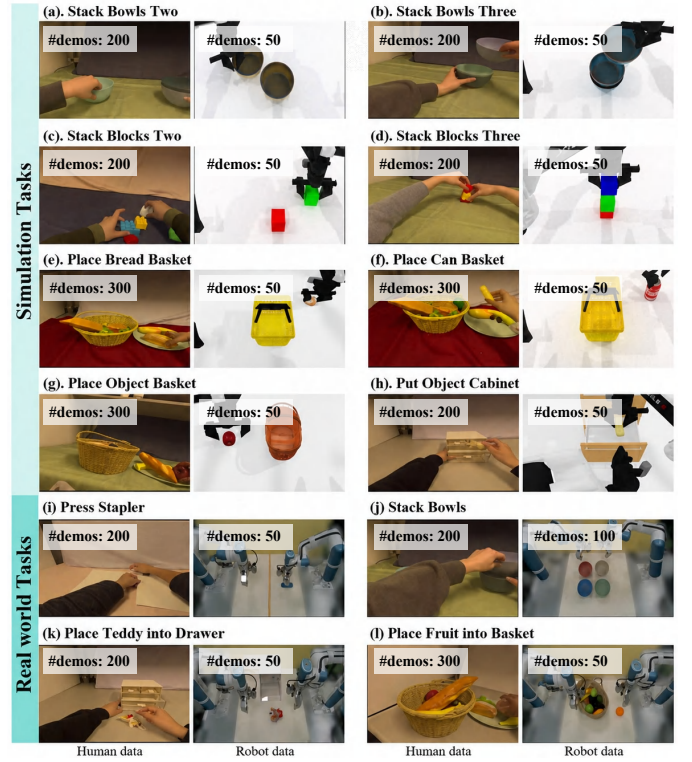


Fig. 3: Representative tasks in simulation and real-world settings, with the number of collected demonstrations indicated for each task.

stages: the visual encoder is warmed up for 20 epochs using Eq. (6), further aligned across human and robot domains for 260 epochs using Eq. (7), and the visuomotor encoder is then trained for 500 epochs with the latent objective in Eq. (11). All training is performed on a single NVIDIA RTX 5090 GPU.

3) *VLA Post-Training*: We adopt $\pi_{0.5}$ [6] as the VLA backbone and initialize all models from the same pretrained checkpoint. To ensure a fair comparison, we use identical optimization settings across all methods. Each model is post-trained with AdamW on two NVIDIA H100 GPUs using a batch size of 64 and a learning rate of 2.5×10^{-5} . We use only the external camera view for policy training and deployment. Models are trained for 20k optimization steps in simulation and 40k steps in the real-world experiments.

IV. EXPERIMENTS AND RESULTS

To evaluate the effectiveness of ReWeight, we conduct experiments in both simulation and real-world environments. We compare ReWeight with robot-only training and naive human-robot data mixing, assessing VLA policy performance under clean and randomized settings across eight simulated and four real-world tasks.

A. Simulation Settings

RoboTwin 2.0 [33] benchmark has been used to evaluate ReWeight across eight bimanual manipulation tasks, including four stacking, three pick-and-place, and one drawer-

TABLE I: Comparison of task success rates for models trained with different robot-human data composition strategies (*refer to Sec. IV-B1 for details*) under clean and randomized simulation settings. Success rates are computed over 100 trials per task.

Tasks	Robot-Only	Mixing Data (All Robot Data + Retrieved Human Data)					
		All _{1.0}	Random _{1.0}	Retrieval _{0.4}	Retrieval _{0.7}	Retrieval _{1.0}	ReWeight (ours)
(a). Clean settings							
Stack Bowls Two	0.88	0.87	0.88	0.89	0.87	<u>0.95</u>	0.96
Stack Bowls Three	0.57	0.62	0.63	0.61	0.67	<u>0.71</u>	0.78
Stack Blocks Two	0.30	0.51	0.44	0.47	0.68	<u>0.71</u>	0.74
Stack Blocks Three	0.14	0.16	0.22	0.26	0.24	<u>0.29</u>	0.30
Place Bread Basket	0.19	0.20	0.22	0.17	0.17	<u>0.25</u>	0.31
Place Can Basket	0.29	0.30	0.31	0.29	<u>0.33</u>	0.32	0.37
Place Object Basket	0.34	0.44	0.38	0.47	<u>0.48</u>	0.45	0.51
Put Object Cabinet	0.42	0.37	0.44	0.44	0.40	<u>0.51</u>	0.55
Average	0.39	0.43	0.44	0.45	0.48	<u>0.52</u>	0.57
(b). Randomized settings							
Stack Bowls Two	0.58	0.55	0.61	0.61	0.57	<u>0.65</u>	0.68
Stack Bowls Three	0.48	0.34	0.41	0.46	0.48	<u>0.47</u>	0.48
Stack Blocks Two	0.19	0.10	0.12	0.27	<u>0.28</u>	0.25	0.29
Put Object Cabinet	0.12	0.15	0.13	0.17	<u>0.19</u>	<u>0.22</u>	0.23
Average	0.34	0.29	0.32	0.38	0.38	<u>0.40</u>	0.42

Robot-Only: only use robot data; **Best** and second-best results are shown in bold and underlined, respectively.

manipulation. Fig. 3 shows robot data and the corresponding egocentric human data from EgoDex [7], together with their data sizes. We divide the eight tasks into two groups based on their manipulation characteristics: the four stacking tasks and the remaining four tasks. Following the training procedure in *Section III-D*, we then train a separate visuomotor encoder for each group to retrieve and weight human data for the corresponding robot tasks. A separate $\pi_{0.5}$ [6] model is then post-trained for each group using the combined robot and retrieved human data. Unless otherwise stated, in all the following experiments, we use all available robot data and retrieve an equal size of human data for each task.

B. Simulation Results

1) *Comparison Study*: Table I compares different strategies for incorporating human data into $\pi_{0.5}$ post-training. To validate the effectiveness of incorporating the human data, we compared post-training with robot data alone (“*Robot Only*”) against post-training with a mixture of robot and human data (“*Mixing Data*”). To assess the necessity of data retrieval, we further compared post-training $\pi_{0.5}$ using all available human data (“*All_{1.0}*”) with using an equally sized, randomly sampled human data (“*Random_{1.0}*”); all human samples in these two variants were assigned a fixed weight of 1.0. To evaluate the benefit of sample weighting, we used ReWeight to retrieve human data with high similarity and post-trained the model with uniform sample weights of 0.4, 0.7, and 1.0, denoted as “*Retrieval_{0.4}*”, “*Retrieval_{0.7}*”, and “*Retrieval_{1.0}*”, respectively.

Simply incorporating human data improves performance under the clean setting where “*All_{1.0}*” and “*Random_{1.0}*” increase the average success rate over “*Robot-Only*” by 4% and 5%, respectively. However, these gains do not transfer to the randomized setting. Under the randomized setting, they instead reduce performance by 5% and 2% percentage points.

This contrast suggests that indiscriminately mixing human and robot data may introduce irrelevant or mismatched samples, leading the model to overfit to the clean training distribution rather than generalize to unseen environmental variations. These findings support our hypothesis that more human data will not automatically result into better policy. The relevance of the selected demonstrations is critical.

The variants trained with human data selected by the proposed demonstration-level retrieval method consistently outperform both “*All_{1.0}*” and “*Random_{1.0}*” under both clean and randomized settings. In the randomized settings, “*Retrieval_{0.4}*” outperforms “*Random_{1.0}*” by 6%, demonstrating the effectiveness of the proposed retrieval method. Among the retrieval variants with uniform weights, increasing the human-data weight from 0.4 (“*Retrieval_{0.4}*”) to 1.0 (“*Retrieval_{1.0}*”) improves the average success rate by 7% in the clean setting and 2% in the randomized setting. ReWeight further assigns a sample-specific weight to each retrieved sample according to Eq. (14). Compared with the strongest uniform-weight variant, “*Retrieval_{1.0}*”, ReWeight achieves further gains of 5% and 2% in the clean and randomized settings, respectively. Overall, these results demonstrate the benefits of combining demonstration-level retrieval with sample-level weighting.

2) *Ablation Study*: We conducted ablation studies on two key design choices of the visuomotor encoder: (i) incorporating future action chunks into the learned representation and (ii) imposing a temporal mismatch penalty when solving the OT problem for learning in Eq. (8) and retrieval with Eq. (4). The results are reported in Table II. In “*w/o Action Info.*”, we remove the action input and rely solely on the learned visual features for cross-embodiment matching. To assess the effectiveness of the temporal penalty $\mathbf{Q}$ in the cost matrix defined in Eq. (9), we additionally evaluate a variant without $\mathbf{Q}$ (“*w/o Temporal Penalty*”). We evaluate the

TABLE II: Cross-embodiment alignment performance of visuomotor encoder under different ablation settings.

Variants	Stack Bowls Two		Stack Bowls Three		Stack Blocks Two		Stack Blocks Three		Average	
	Stage Acc.	KF Dist.	Stage Acc.	KF Dist.	Stage Acc.	KF Dist.	Stage Acc.	KF Dist.	Stage Acc. $\uparrow$	KF Dist. $\downarrow$
w/o Action Info.	0.84	11.2	0.84	5.9	0.80	11.2	0.85	11.6	0.83	10.0
w/o Temporal Penalty	0.86	8.5	0.86	4.6	0.84	9.2	0.88	9.5	0.86	8.0
Visuomotor Enc. (Ours)	0.95	0.8	0.93	2.6	0.91	5.6	0.91	7.5	0.93	4.1

“w/o Action Info.” refers to Visual Encoder; “w/o Temporal Penalty” refers to without temporal mismatch penalty Q in Eq. (9).

Stage Acc.: Stage accuracy; KF Dist.: Key frame distance; Visuomotor Enc.: Visuomotor Encoder.

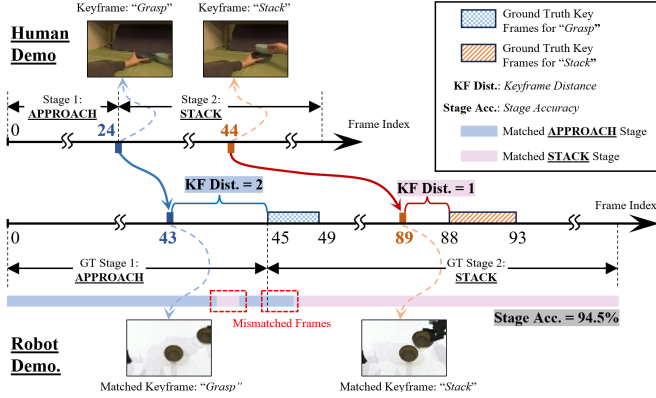


Fig. 4: Illustration of cross-embodiment key frame correspondence and the calculation of Stage Accuracy (*Stage Acc.*) and Keyframe Distance (*KF Dist.*).

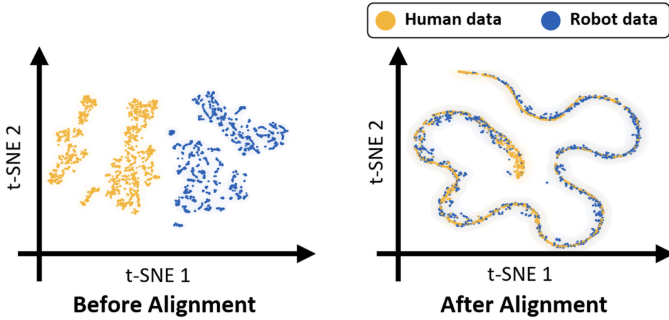


Fig. 5: t-SNE visualization of human and robot visual representations before and after cross-embodiment alignment.

encoder variants on four simulation tasks: *Stack Bowls Two*, *Stack Bowls Three*, *Stack Blocks Two*, and *Stack Blocks Three*. Cross-embodiment correspondence quality is assessed using two metrics: Stage Accuracy and Keyframe Distance. Fig. 4 illustrates a representative key frame correspondence between a human demonstration and its matched robot demonstration, together with the definitions of both metrics. Specifically, for each human demonstration, we computed its Wasserstein distance to every robot demonstration from the same task and selected the robot demonstration with the smallest distance as its match. Stage accuracy (*Stage Acc.*) and key frame distance (*KF Dist.*) are then computed for each matched human-robot pair. The values reported in Table II are averaged over all human demonstrations across the four tasks.

TABLE III: Success rates under different values of the sample-weight calibration parameter α in Eq. (14).

Values	Stack Bowls Three	Stack Blocks Two	Place Object Basket	Average
Robot Only	0.57	0.30	0.34	0.40
$\alpha = 0.1$	0.77	0.71	0.46	0.65
$\alpha = 0.3$	0.77	0.70	0.47	0.65
$\alpha = 0.5$	0.78	0.74	0.51	0.68
$\alpha = 0.7$	0.74	0.72	0.46	0.64
$\alpha = 1.0$	0.71	0.71	0.45	0.62

Robot Only: no human data for training; $\alpha = 1.0$: unweighted human data.

Notably, removing the action input (“w/o Action Info.”) consistently degrades performance across all four simulation tasks, reducing the average stage accuracy from 0.93 to 0.83 and increasing the average keyframe distance from 4.1 to 10.0 frames. Fig. 5 visualizes the features learned by the vision encoder before and after cross-embodiment alignment using t-SNE. Before alignment, the human and robot features form clearly separated domains, whereas after alignment, they are better integrated into a shared feature space. This confirms the effectiveness of the visual alignment strategy for cross-embodiment similarity estimation. However, the inferior performance of the visual-only variant shows that visual alignment alone is insufficient for reliable retrieval, highlighting the critical role of future action information. Removing the temporal penalty (“w/o Temporal Penalty”) also consistently degrades both metrics, although to a lesser extent, with a 7% drop in stage accuracy and a 3.9-frame increase in keyframe distance. These results show that action information is essential for capturing behaviorally meaningful cross-embodiment discrepancies, while the temporal prior promotes temporally consistent human-robot correspondences.

We further investigate the sensitivity of ReWeight to the minimum human-sample weight α in Eq. (14) by evaluating task success rates on three representative simulation tasks, including *Stacking Bowls*, *Stacking Blocks*, and *Placing Object Basket*. A smaller α more strongly suppresses human samples with large cross-embodiment discrepancies, whereas a larger α allows all retrieved samples to contribute more to policy optimization. As shown in Table III, $\alpha = 0.5$ achieves the best downstream performance by maintaining a reasonable minimum contribution from each retrieved human sample while still differentiating samples according to their cross-embodiment relevance.

TABLE IV: Performance comparison under real-world perturbations, including lighting changes and visual distractors.

Tasks	Robot Only		Naive Mixing		ReWeight (Ours)	
	Lighting	Distractors	Lighting	Distractors	Lighting	Distractors
Stack Bowls	2/10	2/10	4/10	2/10	7/10	5/10
Place Teddy Drawer	3/10	5/10	6/10	4/10	5/10	5/10
Pick Fruits Basket	5/10	4/10	5/10	7/10	7/10	9/10
Press Stapler	2/10	0/10	3/10	2/10	5/10	5/10
Overall	12/40 (30%)	11/40 (28%)	18/40 (45%)	15/40 (38%)	24/40 (60%)	24/40 (60%)

Robot-Only: only use robot data; Naive Mixing: randomly sampled human data + all robot data; **Best** results are shown in bold.

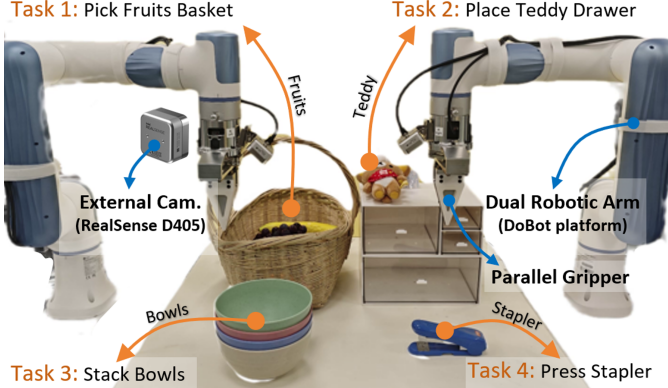


Fig. 6: Overview of the real-world experimental setup and the four evaluation tasks.

C. Realworld Evaluation

As shown in Fig. 6, we evaluate ReWeight on a dual-arm DoBot platform across four real-world tasks: *Stack Bowls*; *Place Teddy Drawer* (Place Teddy bear into a drawer and close it); *Pick Fruit Basket* (Pick fruits and place them into a basket); and *Press Stapler*. Robot data were collected via teleoperation. The sizes of the robot data and the corresponding human-data pools from EgoDex [7] for retrieval are presented in Fig. 3.

Following the simulation setup, we compare ReWeight against two baselines: “*Robot Only*”, which post-trains $\pi_{0.5}$ using only robot data, and “*Naive Mixing*”, which randomly samples and incorporates human data during post-training. All models were evaluated under both clean and randomized settings. Under the clean setting, we conduct 20 trials per task, varying only the initial object positions while keeping all other environmental conditions unchanged. For *Pick Fruit into Basket*, the evaluation additionally includes fruit categories present in EgoDex but absent from the robot data. Under the randomized setting, we assess robustness to visual variations using two controlled real-world perturbations: lighting variations and unseen visual distractors. For each perturbation, we conducted 10 trials per task.

D. Real-world Results

As shown in Fig. 7, ReWeight consistently outperforms both “*Robot Only*” and “*Naive Mixing*” across all four real-world tasks, improving the average success rate by 28.8 and 13.8

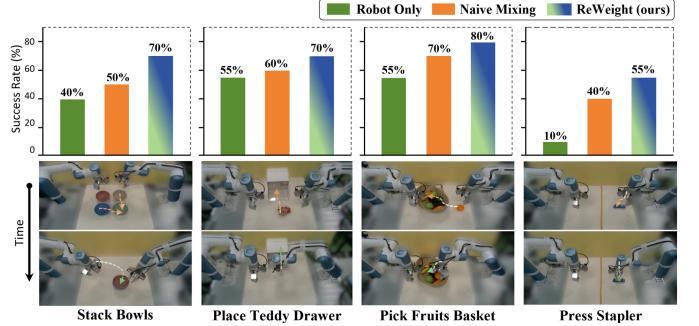


Fig. 7: Performance comparison across four real-world tasks under the clean setting, with 20 trials conducted per task.

percentage points, respectively. This consistent improvement demonstrates that selectively retrieving and weighting egocentric human demonstrations can effectively enhance real-world robot manipulation performance. Under the randomized settings, as reported in Table IV, ReWeight also exhibits greater robustness to environmental perturbations. Across all four tasks, ReWeight achieves an overall success rate of 60.0% (48/80), compared with 41.3% (33/80) for “*Naive Mixing*” and 28.8% (23/80) for “*Robot Only*”, corresponding to improvements of 18.7 and 31.2 percentage points, respectively. The improvement is particularly pronounced on *Pick Fruits Basket* task, where ReWeight succeeds in 16/20 trials, compared with 12/20 and 9/20 trials for “*Naive Mixing*” and “*Robot Only*”, respectively. These results demonstrate that human-data retrieval and sample weighting improve robustness to visual variations beyond naive mixing of human and robot demonstrations. Moreover, ReWeight successfully manipulates object categories such as bananas and grapes, which are absent from the robot teleoperation data but present in the egocentric human demonstrations, indicating improved object-level generalization.

V. CONCLUSION

This work introduced ReWeight, a framework for effectively incorporating egocentric human demonstrations into VLA post-training. Rather than naively mixing human and robot data, ReWeight determines both which human demonstrations to transfer and how strongly each sample should contribute to policy optimization. It learns a cross-embodiment visuomotor representation to quantify human–robot discrepancies, retrieve

relevant demonstrations, and assign discrepancy-aware sample weights. In simulation, ReWeight improves the average success rate of post-trained $\pi_{0.5}$ from 39% with robot-only data and 44% with randomly mixed human–robot data to 57%. In real-world experiments, it achieves 68.8%, outperforming the two baselines by 28.8% and 13.8%, respectively. Under lighting changes and visual distractors, ReWeight maintains a 60.0% success rate, compared with 28.8% and 41.3% for the baselines. These results demonstrate the effectiveness and robustness of ReWeight for leveraging human demonstrations in VLA post-training.

REFERENCES

- [1] A. O’Neill, A. Rehman, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain *et al.*, “Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0,” in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 6892–6903.
- [2] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi *et al.*, “Openvla: An open-source vision-language-action model,” *arXiv preprint arXiv:2406.09246*, 2024.
- [3] J. Zhang, Z. Jin, Z. Zhao, and C. Yang, “A novel robotic skill learning approach for assembly task with dynamical system and broad learning,” *IEEE Transactions on Industrial Electronics*, vol. 72, no. 9, pp. 9304–9313, 2025.
- [4] W. Si, N. Wang, R. Harris, and C. Yang, “Deep multimodal imitation learning-based framework for robot-assisted medical examination,” *IEEE Transactions on Industrial Electronics*, 2025.
- [5] H. Huang, C. Guo, Z. Xie, J. Guo, Z. An, and P. Chen, “Convex-guided elastic dynamical systems: A unified framework for stable and collision-free motion generation,” *IEEE Transactions on Industrial Electronics*, 2026.
- [6] K. Black, N. Brown, J. Darpinian, K. Dhabalia, D. Driess, A. Esmail, M. R. Equi, C. Finn, N. Fusai, M. Y. Galliker, D. Ghosh, L. Groom, K. Hausman, b. ichter, S. Jakubczak, T. Jones, L. Ke, D. LeBlanc, S. Levine, A. Li-Bell, M. Mothukuri, S. Nair, K. Pertsch, A. Z. Ren, L. X. Shi, L. Smith, J. T. Springenberg, K. Stachowicz, J. Tanner, Q. Vuong, H. Walke, A. Walling, H. Wang, L. Yu, and U. Zhilinsky, “ $\pi_{0.5}$: a vision-language-action model with open-world generalization,” in *Proceedings of The 9th Conference on Robot Learning*, vol. 305. PMLR, 27–30 Sep 2025, pp. 17–40.
- [7] R. Hoque, P. Huang, D. Yoon, J. Zhang *et al.*, “Egodex: Learning dexterous manipulation from large-scale egocentric video,” in *International Conference on Learning Representations*, vol. 2026, 2026, pp. 4218–4237.
- [8] Q. Li, Y. Deng, Y. Liang, L. Luo, L. Zhou, C. Yao, L. Zeng, Z. Feng, H. Liang, S. Xu *et al.*, “Scalable vision-language-action model pretraining for robotic manipulation with real-life human activity videos,” *arXiv preprint arXiv:2510.21571*, 2025.
- [9] R. Yang, Q. Yu, Y. Wu, R. Yan, B. Li, A.-C. Cheng, X. Zou, Y. Fang, X. Cheng, R.-Z. Qiu *et al.*, “Egovla: Learning vision-language-action models from egocentric human videos,” *arXiv preprint arXiv:2507.12440*, 2025.
- [10] H. Luo, Y. Feng, W. Zhang, S. Zheng, Y. Wang, H. Yuan, J. Liu, C. Xu, Q. Jin, and Z. Lu, “Being-h0: vision-language-action pretraining from large-scale human videos,” *arXiv preprint arXiv:2507.15597*, 2025.
- [11] S. Kareer, D. Patel, R. Punamiya, P. Mathur, S. Cheng, C. Wang, J. Hoffman, and D. Xu, “Egomimic: Scaling imitation learning via egocentric video,” in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025, pp. 13 226–13 233.
- [12] S. Kareer, K. Pertsch, J. Darpinian, J. Hoffman, D. Xu, S. Levine, C. Finn, and S. Nair, “Egomimic: Scaling imitation learning via vision-language-action models,” *arXiv preprint arXiv:2512.22414*, 2025.
- [13] H. Luo, Y. Wang, W. Zhang, S. Zheng, Z. Xi, C. Xu, H. Xu, H. Yuan, C. Zhang, Y. Wang *et al.*, “Being-h0. 5: Scaling human-centric robot learning for cross-embodiment generalization,” *arXiv preprint arXiv:2601.12993*, 2026.
- [14] Y. Wang, S. Zheng, H. Luo, W. Zhang, H. Yuan, C. Xu, H. Xu, Y. Feng, M. Yu, Z. Kang *et al.*, “Rethinking visual-language-action model scaling: Alignment, mixture, and regularization,” *arXiv preprint arXiv:2602.09722*, 2026.
- [15] R. Punamiya, S. Kareer, Z. Liu, J. Citron, R.-Z. Qiu, X. Cai, A. Gavryushin, J. Chen, D. Liconti, L. Y. Zhu *et al.*, “Egoverse: An egocentric human dataset for robot learning from around the world,” *arXiv preprint arXiv:2604.07607*, 2026.
- [16] M. Hong, A. Liang, K. Kim, H. Rajaprakash, J. Thomason, E. Bıyık, and J. Zhang, “Hand me the data: Fast robot adaptation via hand path retrieval,” *arXiv preprint arXiv:2505.20455*, 2025.
- [17] Y. Zhu and F. Feng, “Let me show you: Learning by retrieving from egocentric video for robotic manipulation,” in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2025, pp. 9824–9829.
- [18] N. Lin, T. Ohkawa, S. Chen, R. Wen, Z. Li, L. Huang, Z. Zhu, Y. Bao, Y. Li, M. Cai *et al.*, “Simdex: Mining similar egocentric videos for cross-embodiment dexterous manipulation,” *arXiv preprint arXiv:2608.04196*, 2026.
- [19] A. Xie, R. Chand, D. Sadigh, and J. Hejna, “Data retrieval with importance weights for few-shot imitation learning,” *arXiv preprint arXiv:2509.01657*, 2025.
- [20] W. Zhang, H. Luo, S. Zheng, Y. Feng, H. Xu, Z. Xi, C. Xu, H. Yuan, and Z. Lu, “Conservative offline robot policy learning via posterior-transition reweighting,” *arXiv preprint arXiv:2603.16542*, 2026.
- [21] M. Cuturi, “Sinkhorn distances: Lightspeed computation of optimal transport,” *Advances in neural information processing systems*, vol. 26, 2013.
- [22] Z. K. Weng, M. L. Elwin, and H. Liu, “Levr: A modular vr teleoperation framework for imitation learning in dexterous manipulation,” *arXiv preprint arXiv:2509.14349*, 2025.
- [23] T. Johannink, S. Bahl, A. Nair, J. Luo, A. Kumar, M. Loskyll, J. A. Ojea, E. Solowjow, and S. Levine, “Residual reinforcement learning for robot control,” in *2019 international conference on robotics and automation (ICRA)*. IEEE, 2019, pp. 6023–6029.
- [24] X. B. Peng, P. Abbeel, S. Levine, and M. Van de Panne, “Deepmimic: Example-guided deep reinforcement learning of physics-based character skills,” *ACM Transactions On Graphics (TOG)*, vol. 37, no. 4, pp. 1–14, 2018.
- [25] Q. Liao, T. E. Truong, X. Huang, Y. Gao, G. Tevet, K. Sreenath, and C. K. Liu, “Beyondmimic: From motion tracking to versatile humanoid control via guided diffusion,” *Science Robotics*, vol. 11, no. 117, p. eadx8924, 2026.
- [26] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, “Unsupervised learning of visual features by contrasting cluster assignments,” *Advances in neural information processing systems*, vol. 33, pp. 9912–9924, 2020.
- [27] P. Sermanet, C. Lynch, Y. Chebotar, J. Hsu, E. Jang, S. Schaal, S. Levine, and G. Brain, “Time-contrastive networks: Self-supervised learning from video,” in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 1134–1141.
- [28] D. Dwibedi, Y. Aytar, J. Tompson, P. Sermanet, and A. Zisserman, “Temporal cycle-consistency learning,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019, pp. 1801–1810.
- [29] M. Cuturi and M. Blondel, “Soft-dtw: a differentiable loss function for time-series,” in *International conference on machine learning*. PMLR, 2017, pp. 894–903.
- [30] S. Zhao, J. Song, and S. Ermon, “Infovae: Balancing learning and inference in variational autoencoders,” in *Proceedings of the aaai conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 5885–5892.
- [31] S. Haldar, V. Mathur, D. Yarats, and L. Pinto, “Watch and match: Supercharging imitation with regularized optimal transport,” in *Conference on Robot Learning*. PMLR, 2023, pp. 32–43.
- [32] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” *arXiv preprint arXiv:1707.06347*, 2017.
- [33] T. Chen, Z. Chen, B. Chen, Z. Cai, Y. Liu, Z. Li, Q. Liang, 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.