INTERACT: GROUNDING LATENT ACTIONS THROUGH CAMERA–PHYSICS INTERVENTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Latent action models learn transitions from video, but visual changes reflect both physical interaction and camera motion. Temporal consistency does not guarantee camera invariance. We introduce InterACT, which adapts a pretrained latent action model through crossed camera and physical interventions. Each quartet pairs two interactions with two camera trajectories. A bounded residual aligns views, a frozen reference anchors physical distance scale, and separate gradient paths train the correction and encoder. InterACT achieves lower camera ratios and action errors and higher outcome gains than video continuation and an existing ego-supervised adaptation across three controlled domains. On unseen camera trajectories, it lowers the camera-to-physical distance ratio from 3.08 to 0.71 while retaining near-reference physical scale, and reduces outcome prediction error by 32.15% relative to a state-only probe. Controlled cross-view Recall@5 increases from 3.39% to 29.43%. A separate comparison with a supervised baseline reveals a trade-off: the full method better preserves reference scale, while supervised adaptation yields higher outcome gains. For downstream policy learning, the frozen tokenizer yields 88.6% success with π_0 on MetaWorld MT50 and 76.11% with ABot-M0.5 on RoboCasa365 atomic-seen tasks, exceeding the reported references of 85.0% and 70.6%, respectively.

1 INTRODUCTION

Latent action models learn transition representations from videos without requiring executable action labels (Ye et al., 2024; Bruce et al., 2024; Gao et al., 2025). These representations support policy learning from human and robot videos (Bu et al., 2025; Jiang et al., 2026a) and connect video prediction with action generation (Bi et al., 2026; Chen et al., 2026). This allows policy learning to use videos beyond action-annotated demonstrations, but the value of these targets depends on what they encode. For control, latent targets should retain information about physical interaction while reducing variation introduced by the camera.

Observed transitions combine physical interaction with camera motion. The same push produces different image changes under fixed and moving cameras, while distinct pushes may appear similar when camera motion compensates for object motion. A latent that supports video reconstruction can therefore assign different policy targets to views of the same interaction. Useful supervision should align these views while preserving differences between interactions. Temporal composition and reversal organize transitions over time (Tang et al., 2026a;b; Wei et al., 2026a), but camera-dependent embedding differences can satisfy both identities exactly. Temporal consistency therefore does not guarantee camera invariance. Existing work also addresses visual variation through action-guided alignment (Jeong et al., 2026) and action-centric learning with latent calibration (Wei et al., 2026b). We assess alignment together with physical scale: constant representations align all views, while a camera-to-physical distance ratio can decrease through expanded physical distances.

We use *crossed camera and physical interventions* to make these requirements explicit. From a shared initial state in a simulator, we generate two physical trajectories and render each under two camera trajectories. The resulting quartet provides matched comparisons: changing the camera while holding the interaction fixed reveals camera variation, while changing the interaction under the same camera reveals physical differences (Figure 1). These comparisons supplement ordinary videos with explicit targets for reducing camera variation and preserving physical differences.

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

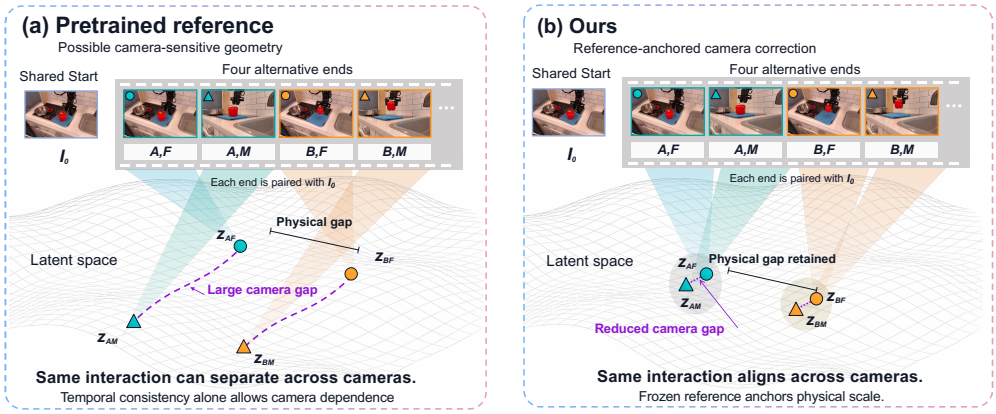


Figure 1: **Separating camera variation from physical change.** Interactions A and B are observed by fixed (F) and moving (M) cameras. Adaptation brings together the two views of each interaction while preserving the difference between interactions.

Our method, InterACT, learns a bounded residual correction from these comparisons. A frozen copy of the initial encoder normalizes camera distances and constrains changes in physical scale. Alignment losses train the correction, while physical constraints and action and outcome supervision train the encoder through a separate gradient path. Reconstruction and temporal learning continue on ordinary videos. The tokenizer takes only RGB observation pairs; intervention identities and interaction labels are used for training supervision.

Controlled evaluation first tests camera alignment, physical scale, and interaction information. On unseen camera trajectories, InterACT reduces the camera-to-physical ratio from 3.08 to 0.71, retains physical scale at 0.97 of the reference on average, and achieves a 32.15% outcome prediction gain over a state-only probe. Across all three domains, it achieves lower camera ratios and action errors and higher outcome gains than video continuation and an existing ego-supervised adaptation. Controlled retrieval also improves. A separate supervised comparison shows a trade-off: the full method better preserves reference scale, while the supervised baseline achieves higher outcome gains. The frozen tokenizer then supplies policy targets for π_0 on MetaWorld MT50 and ABot-M0.5 on RoboCasa365. The adapted targets yield 88.6% success on MetaWorld, compared with the reported 85.0% latent-action reference (Tang et al., 2026a). On RoboCasa365 atomic-seen tasks, ABot-M0.5 reaches 76.11%, compared with its reported 70.6% reference. Both policies use the same frozen tokenizer.

Our contributions are as follows:

- We characterize camera dependence compatible with temporal composition and reversal, and construct crossed interventions that distinguish camera variation from physical differences.
- We introduce an adaptation framework using a bounded residual, a frozen physical reference, constraints on physical scale, and separate gradient paths for the correction and encoder.
- We compare representation geometry and prediction across video and supervised adaptations, examine scale–prediction trade-offs, and evaluate retrieval and downstream use on MetaWorld MT50 and RoboCasa365.

2 Related Work

2.1 Learning latent actions from video

Latent action models infer transition variables from observation sequences, providing supervision for generative environments and policy pretraining without requiring an executable action label for every video (Bruce et al., 2024; Ye et al., 2024). Continuous representations also support adaptable world models (Gao et al., 2025). Recent work structures these variables through temporal composition and reversal (Tang et al., 2026a;b), or aggregates short transitions into longer-horizon latent skills (Kim

et al., 2026). These approaches organize transitions over time. We study the camera dependence that can remain even when these temporal relations are consistent.

2.2 Grounding and aligning latent actions

Analytical work studies how controllable changes and external noise affect latent action learning (Zhang et al., 2025). To reduce dependence on visual appearance, recent approaches separate transition information from scene content Zhang et al. (2026), contrast action-related video pairs (Dai et al., 2026), or introduce optical-flow constraints and semantic, dynamic, and kinematic structure (Bu et al., 2026; Liu et al., 2026). Multi-view and action-centric learning further use action-guided alignment (Jeong et al., 2026) and combinations of reconstruction, contrastive learning, and latent calibration (Wei et al., 2026b). Temporal changes in frozen video features can also provide a reference for aligning latent action semantics across contexts (Jiang et al., 2026b). We use crossed camera–physics pairs to align views of the same interaction while preserving differences between interactions. A frozen physical reference lets us constrain distance scale during alignment.

2.3 Latent actions in policy learning

Transitions learned from video provide intermediate targets for policy pretraining and transfer across embodiments (Ye et al., 2024; Bu et al., 2025). Predictive embeddings can also be aligned with inverse dynamics and available action labels to learn from heterogeneous human videos (Luo et al., 2026). Latent pretraining from egocentric video has also been used for humanoid locomotion and manipulation (Jiang et al., 2026a). Downstream policies jointly predict latent transitions and robot actions (Tang et al., 2026a), couple video prediction with action generation (Guo et al., 2024; Chen et al., 2026; Bi et al., 2026), or extract motion-aligned latent dynamics from predicted futures to condition control (Shen et al., 2026). We retain established policy interfaces and adapt the tokenizer that supplies their latent targets. Our evaluation first measures camera robustness and physical retention, then tests the frozen tokenizer in downstream policy learning.

3 Method

Our method adapts a pretrained transition encoder with ordinary videos and controlled intervention quartets (Figure 2). The adaptation aligns views of the same interaction while preserving differences between interactions; the frozen tokenizer then supplies auxiliary targets for policy learning.

3.1 Crossed Camera and Physical Interventions

Each quartet starts from a shared simulator state. Two physical trajectories, indexed by $j \in \{0, 1\}$, are each rendered under two camera trajectories, indexed by $i \in \{0, 1\}$. Their endpoints define four observation pairs $(I_{0,i,j}, I_{\Delta,i,j})$ over a frame interval Δ. Changing i holds the interaction fixed; changing j holds the camera fixed. The controlled data pair contact and non-contact interactions with fixed and translating cameras.

An encoder f_θ, initialized from a pretrained latent action model, maps each observation pair to a raw latent. A residual module r_ϕ corrects this representation:

$$u_{ij} = f_\theta(I_{0,i,j}, I_{\Delta,i,j}), \qquad z_{ij} = u_{ij} + r_\phi(u_{ij}) \in \mathbb{R}^{K\times H}. \tag{1}$$

Here K is the number of latent slots and H their width. A frozen copy of the initial encoder, f_{θ_0}, extracts reference latents $\bar{u}_{ij}$ from the same pairs. Both encoders take only RGB images; indices and interaction labels supervise training after encoding. This allows target extraction from ordinary demonstration videos. Appendix A.1 gives the architecture and training settings.

3.2 Cross-View Alignment and Physical Preservation

Latent magnitude is measured by the root mean square (RMS) over all KH coordinates. We use $n_\epsilon(x) = \sqrt{\|x\|^2_{\mathrm{rms}} + \epsilon^2}$ to keep normalization well-defined at zero, and bound the correction by $\|r_\phi(u)\|_{\mathrm{rms}} \le \alpha n_\epsilon(u)$. Here $\epsilon > 0$ provides smoothing and α controls the correction relative to its input. The encoder remains trainable. Appendix A.2 gives the clipping rule.

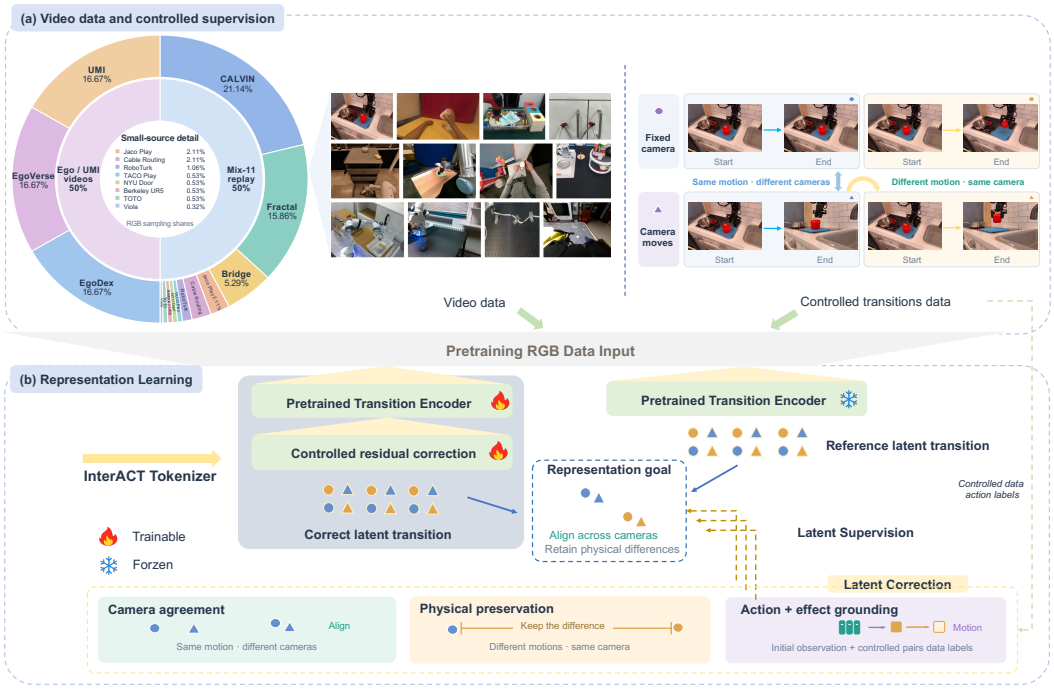


Figure 2: **Training overview.** (a) RGB video sources and sampling shares, alongside crossed camera–physics quartets. (b) Alignment trains the residual, while physical preservation and interaction supervision adapt the encoder. A frozen encoder supplies the physical reference. The scenes illustrate the pairing design; quantitative tests use rendered controlled transitions.

Cross-view alignment. For each quartet, we compute camera differences, physical differences, and the corresponding frozen physical references:

$$\delta_{\mathrm{cam},j} = z_{1j} - z_{0j}, \qquad \delta_{\mathrm{phys},i} = z_{i1} - z_{i0}, \qquad \bar{\delta}_{\mathrm{phys},i} = \bar{u}_{i1} - \bar{u}_{i0}. \tag{2}$$

Their mean magnitudes are $D_{\mathrm{cam}} = \frac{1}{2}\sum_j n_\epsilon(\delta_{\mathrm{cam},j})$ and $D_{\mathrm{phys}} = \frac{1}{2}\sum_i n_\epsilon(\delta_{\mathrm{phys},i})$; $\bar{D}_{\mathrm{phys}}$ averages the frozen differences in the same way. We normalize camera variation by this reference and minimize

$$\mathcal{L}_{\mathrm{cam}} = \mathbb{E}_{\mathcal{Q}}\left[\max\{0,\, D_{\mathrm{cam}}/\bar{D}_{\mathrm{phys}} - \tau\}^2\right]. \tag{3}$$

Here $\mathcal{Q}$ is the training quartet set and τ is the allowed camera variation relative to the reference scale. The loss penalizes only the excess. Using a frozen denominator prevents expansion of the adapted physical distances from lowering this loss on its own.

Physical preservation. For each camera, the ratio $\rho_i = n_\epsilon(\delta_{\mathrm{phys},i})/n_\epsilon(\bar{\delta}_{\mathrm{phys},i})$ measures physical scale relative to the frozen reference. We penalize values outside $[\rho_{\min}, \rho_{\max}]$:

$$\mathcal{L}_{\mathrm{scale}} = \mathbb{E}_{\mathcal{Q},i}\left[\max\{0,\, \log(\rho_{\min}/\rho_i),\, \log(\rho_i/\rho_{\max})\}^2\right]. \tag{4}$$

This loss is zero inside the band and measures contraction or expansion outside it on a log scale. Additional terms constrain the residual, preserve reference directions, match physical differences across views, and regularize variance (Appendix A.2).

3.3 Interaction Supervision and Gradient Routing

Action and outcome prediction encourage the corrected latent to retain interaction information. The action head g_a uses the full latent, while the outcome head g_e also receives initial-image features from g_s:

$$\hat{a}_{ij} = g_a(z_{ij}), \qquad \hat{e}_{ij} = g_e(g_s(I_{0,i,j}), z_{ij}). \tag{5}$$

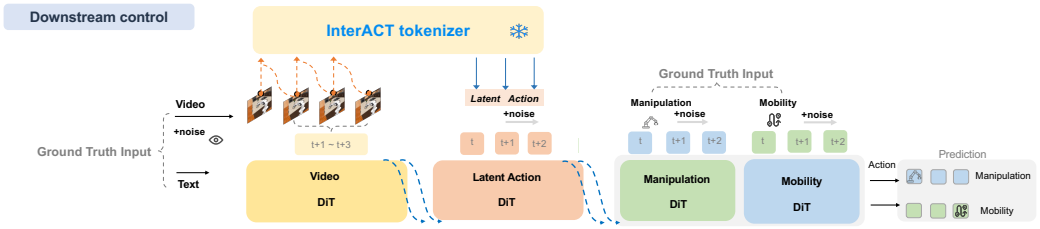


Figure 3: **Downstream control with the frozen tokenizer.** Demonstration frame pairs provide continuous latent targets. ABot-M0.5 retains its video, latent, and action streams and their prediction order. At inference, the policy conditions on current inputs and executes the generated robot actions.

The targets a_j and e_j are shared across cameras for the same physical trajectory. Outcomes describe object displacement, final orientation, and contact. We minimize

$$\mathcal{L}_{\text{sup}} = \mathbb{E}_{\mathcal{Q},i,j}[\ell_{\text{reg}}(\hat{a}_{ij}, a_j) + \ell_{\text{reg}}(\omega_e \odot \hat{e}_{ij}, \omega_e \odot e_j)] \,, \tag{6}$$

where ℓ_{reg} is coordinate-averaged SmoothL1, ω_e rescales outcome coordinates, and $\odot$ denotes elementwise multiplication. To prevent view matching from directly changing the encoder, we use two paths with identical forward values and different gradients:

$$z_{\text{cam}} = \text{sg}[u] + r_\phi(\text{sg}[u]), \qquad z_{\text{phys}} = u + \text{sg}[r_\phi(u)] \,, \tag{7}$$

where sg stops gradients. Alignment, cross-view action consistency, and the residual guard update the correction through z_{cam}; physical and supervised losses train the encoder and heads through z_{phys}. For action consistency, we detach the head parameters and reference prediction while keeping gradients through the alternate-view latent. Video objectives remain on raw latents, with an additional quantization loss on corrected latents. Appendix A.2 gives the remaining losses, scales, and weights.

3.4 Downstream Control

After adaptation, the frozen tokenizer extracts continuous latent targets from demonstration videos. For view i and action-chunk start t, we compute

$$z_{t,i} = f_{\text{tok}}(I_{t,i}, I_{t+\Delta,i}), \qquad \ell_{t,i} = g_\eta(\text{vec}(z_{t,i})), \tag{8}$$

where $f_{\text{tok}} = f_\theta + r_\phi \circ f_\theta$, Δ is the target horizon, and g_η is a trainable projection. Targets retain their time and camera order and are extracted before codebook lookup.

The π_0 policy jointly generates latent transitions and robot actions with flow matching (Tang et al., 2026a). Let y_s be the clean sequence for a latent or robot-action stream s. Using Gaussian noise ξ_s and a shared noise level σ, we form

$$x_{\sigma,s} = (1-\sigma)y_s + \sigma\xi_s, \qquad v_s^* = \xi_s - y_s. \tag{9}$$

The policy predicts the target velocity v_s^* from the noised streams x_σ and context c, which contains current observations, language, and robot state:

$$\mathcal{L}_{\text{policy}} = \sum_{s\in\mathcal{S}} \lambda_s \, \mathbb{E}\big[\|\hat{v}_{\psi,s}(x_\sigma, \sigma, c) - v_s^*\|_1\big] \,, \tag{10}$$

where $\mathcal{S}$ contains the latent and action streams and ψ denotes policy parameters. The interface retains its stream weights λ_s, interleaved token order, and attention mask. At inference, the streams are jointly denoised from Gaussian noise, and only robot actions are executed.

For ABot-M0.5, we supply the frozen tokenizer's targets to its video–latent–action interface (Chen et al., 2026) (Figure 3):

$$\mathcal{L}_{\text{ABot}} = \lambda_v \mathcal{L}_{\text{video}} + \lambda_l \mathcal{L}_{\text{latent-action}} + \lambda_a \mathcal{L}_{\text{action}}. \tag{11}$$

The weights λ_v, λ_l, and λ_a correspond to video, latent-action, and robot-action prediction. The original losses, weights, and ordered sampling procedure are unchanged. Appendix D details the latent loss and projection. Future demonstration frames provide training supervision; deployment requires no future observations.

Table 1: **Overall representation comparison.** "+" denotes adaptation of ALAM; V-adapt and Ego-super denote video adaptation and egocentric motion supervision. Steps count additional optimizer updates. The lower block compares the full method with a separate supervised baseline at 975 updates. Bold highlights comparisons within each block.

Method	Steps	ID				Camera-OOD				Physics-OOD			
		R_{cp}	M	E_a	G_e (%)	R_{cp}	M	E_a	G_e (%)	R_{cp}	M	E_a	G_e (%)
ALAM	0	2.28	1.00	1.21	−302.08	3.08	1.00	1.44	−503.66	2.62	1.00	1.24	−258.00
+ V-adapt	3.9k	6.27	1.12	1.20	−460.88	8.35	1.05	1.28	−559.95	6.66	1.23	1.21	−396.94
+ V-adapt	7.8k	6.33	1.17	1.19	−413.81	8.58	1.07	1.47	−504.19	6.63	1.29	1.20	−356.57
+ Ego-super	3.9k	3.14	0.94	1.07	−223.33	4.24	0.91	1.65	−391.16	3.47	0.98	1.08	−195.47
+ InterACT	3.9k	**0.40**	1.03	**0.20**	**+65.57**	**0.71**	0.97	**0.34**	**+32.15**	**0.60**	1.04	**0.29**	**+42.41**
+ InterACT	975	0.63	**1.09**	**0.38**	+20.35	0.94	**1.05**	0.61	−9.22	0.86	**1.11**	**0.44**	+16.12
Supervised	975	**0.18**	5.72	0.45	**+77.32**	**0.37**	4.93	**0.60**	**+53.59**	**0.29**	5.73	0.50	**+62.95**

4 EXPERIMENTS

Training data. Training uses two streams: video data and synthetic interventions. The video data combine Open-X-Embodiment, CALVIN, EgoDex, EgoVerse, and UMI, with sampling shares shown in Figure 2(a). The synthetic data consist of simulator-rendered quartets: contact and non-contact interactions are each observed under fixed and translating cameras, yielding four RGB transition pairs. We denote the number of training quartets by N_q. The standard and larger sets contain 1,024 and 8,192 quartets, respectively, with $N_q = 640$ and $N_q = 5{,}120$. The larger set additionally varies object position, orientation, and mass. Appendix A.1 provides the complete splits.

4.1 OVERALL REPRESENTATION COMPARISON

Compared settings. The upper block of Table 1 compares four schemes initialized from the same ALAM weights (Tang et al., 2026a). ALAM provides the unadapted reference. Video adaptation continues the original video objectives with the same RGB sampling, effective RGB batch, optimizer, and learning-rate schedule as our method; its two checkpoints come from one run. Ego-supervised adaptation adds egocentric motion labels, joint encoder–residual training, corrected-latent quantization, and RGB reference-distance regularization. Our full adaptation uses $N_q = 640$ controlled quartets with action and outcome supervision.

Evaluation protocol and metrics. We evaluate in-distribution (ID) transitions and out-of-distribution (OOD) camera and physics shifts, denoted Camera-OOD and Physics-OOD. Each domain contains 192 test quartets from 24 initialization clusters, with disjoint clusters across splits. Each model's probes use the same ID reference-view training data, with regularization selected on ID validation data, and remain fixed for alternate-view and OOD tests. Paired comparisons use cluster-bootstrap intervals. Geometry is measured alongside prediction using the unsmoothed RMS distances from Section 3.2:

$$R_{\text{cp}} = \mathbb{E}[D_{\text{cam}}]/\mathbb{E}[D_{\text{phys}}], \qquad M = \mathbb{E}[D_{\text{phys}}/\bar{D}_{\text{phys}}].$$

The camera-to-physical ratio R_{cp} measures camera variation relative to physical variation, while M measures physical scale relative to the frozen reference. We seek a lower R_{cp} with M close to one. Action error E_a is normalized RMSE, with lower values indicating better cross-view action prediction. Outcome gain $G_e = 100(1 - E_{s,z}/E_s)$ measures the percentage reduction in prediction error from adding the latent to the initial state, where E_s and $E_{s,z}$ are errors from the state-only and state-plus-latent probes. Higher gain is better; negative values indicate higher error than the state-only probe. Outcome probes use simulator state, whereas the training head receives initial-image features. Appendix B provides the details.

Cross-view geometry and prediction. At 3.9k updates, InterACT lowers camera ratios and action errors and increases outcome gains across all three domains relative to the pretrained, video-continuation, and ego-supervised baselines in the upper block of Table 1, while keeping physical

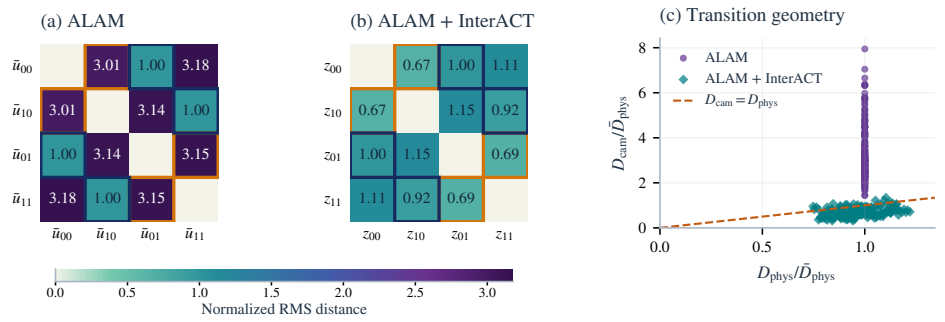


Figure 4: **Camera-OOD latent geometry.** ALAM and ALAM + InterACT (3.9k updates; $N_q = 640$) on 192 quartets. (a,b) Mean pairwise RMS distances normalized by the mean frozen-reference physical distance, with a shared color scale. Subscripts index camera then interaction; orange/blue borders mark camera-only/physical-only pairs. (c) Each point represents one quartet, normalized by its frozen-reference physical distance. The dashed line denotes $D_{\text{cam}} = D_{\text{phys}}$.

Table 2: **Retrieval and reconstruction.** R@1/R@5 denote Recall@1/5. Adaptation uses 3.9k updates and $N_q = 640$. Controlled evaluation uses 384 retrieval candidates and 768 reconstruction targets. Video diagnostics use 128 CALVIN static–gripper windows for retrieval and six Bridge, TACO Play, and Viola examples for reconstruction.

Method	Controlled data					Video data				
	Retrieval (%)		Reconstruction			Retrieval (%)		Reconstruction		
	R@1 ↑	R@5 ↑	PSNR ↑	SSIM ↑	LPIPS ↓	R@1 ↑	R@5 ↑	PSNR ↑	SSIM ↑	LPIPS ↓
ALAM	0.78	3.39	21.16	0.81	0.41	**0.78**	2.73	**26.00**	**0.87**	**0.10**
+ InterACT	**7.55**	**29.43**	**22.13**	**0.87**	**0.32**	0.39	2.73	24.91	0.85	0.11

scale close to the frozen reference. Relative to pretrained ALAM, the Camera-OOD ratio decreases from 3.08 to 0.71 with $M = 0.97$, and action error falls from 1.44 to 0.34. The latent also improves outcome prediction over the state-only probe by 32.15%. Figure 4 illustrates the corresponding geometry: views of the same interaction become closer while distinct interactions remain separated. These advantages hold against video continuation at equal and doubled update counts and against the ego-supervised adaptation. Paired intervals favor InterACT over these controls on R_{cp}, E_a, and G_e in every domain.

Retrieval and reconstruction. Retrieval tests cross-view correspondence without a learned probe, while reconstruction checks visual prediction (Table 2). Recall@1 and Recall@5 report the fractions of queries whose paired transition ranks first or among the five nearest candidates. On controlled data, InterACT improves both diagnostics: Recall@5 rises from 3.39% to 29.43%, and all three reconstruction metrics improve, including PSNR from 21.16 to 22.13. On video data, CALVIN retrieval shows no gain, and pretrained ALAM reconstructs the six fixed examples better. Appendix B provides the sampling protocols and examples.

4.2 Representation Learning Ablations

Supervised adaptation and scale retention. The supervised comparison tests whether stronger prediction coincides with better reference-scale retention. The lower block of Table 1 compares the full method with a supervised baseline after 975 additional updates. InterACT keeps M between 1.05 and 1.11, versus 4.93–5.73 for the baseline. Action error is lower with InterACT on ID and Physics-OOD, but slightly higher on Camera-OOD. The supervised baseline achieves higher outcome gains in all three domains. Its lower camera ratios accompany substantial physical-scale expansion, so the ratio alone does not establish stronger camera invariance. These results distinguish scale retention from predictive performance at this update budget.

Table 3: **Controlled intervention coverage.** Both settings use ALAM + InterACT for 3.9k updates. N_q counts training quartets; the larger set also broadens object position, orientation, and mass.

N_q	ID				Camera-OOD				Physics-OOD			
	$R_{\text{cp}} \downarrow$	$M \approx 1$	$E_a \downarrow$	G_e (%) $\uparrow$	$R_{\text{cp}} \downarrow$	$M \approx 1$	$E_a \downarrow$	G_e (%) $\uparrow$	$R_{\text{cp}} \downarrow$	$M \approx 1$	$E_a \downarrow$	G_e (%) $\uparrow$
640	**0.40**	1.03	**0.20**	$+\mathbf{65.57}$	**0.71**	0.97	**0.34**	$+\mathbf{32.15}$	**0.60**	1.04	**0.29**	$+42.41$
5,120	1.17	1.07	0.39	$+53.43$	1.39	1.02	0.56	$+28.27$	1.37	1.07	0.42	$+\mathbf{51.82}$

Table 4: **Ablations in the action-bottleneck variant of InterACT.** Routing and pairing are evaluated in separate 975-step Camera-OOD comparisons, each with its own Full reference. SG denotes stop-gradient; removing it allows the corresponding loss to update the encoder. Removing correct pairing shuffles camera–interaction correspondences.

Metric	Gradient routing				Transition pairing	
	Full	w/o outcome SG	w/o alignment SG	w/o both SGs	Full	w/o correct pairing
$R_{\text{cp}} \downarrow$	0.88	0.93	0.87	1.04	0.95	0.99
$M \approx 1$	0.93	0.95	13.28	10.68	0.95	1.03
$E_a \downarrow$	0.66	0.63	1.47	1.13	0.52	0.53
G_e (%) $\uparrow$	-5.35	-13.45	-308.49	-144.60	-99.15	-12.11

Intervention coverage. The coverage study tests generalization by increasing N_q from 640 to 5,120 and broadening object position, orientation, and mass variation. The architecture and 3.9k update budget remain fixed. Table 3 shows that Physics-OOD outcome gain rises from 42.41% to 51.82%, while camera ratios and action errors increase across all three domains. Broader coverage therefore helps outcome prediction under the physical shift, but does not improve alignment or action readout. Appendix C provides training curves and paired intervals.

Gradient routing and pairing. Our routing separates correction learning from encoder adaptation. Table 4 tests this choice in the action-bottleneck variant by allowing outcome consistency, alignment, or both to send additional gradients to the encoder, with all other settings fixed within each comparison. Removing the alignment stop-gradient lowers the camera ratio from 0.88 to 0.87, but increases M from 0.93 to 13.28 and action error from 0.66 to 1.47. Removing both stop-gradients also expands physical scale. The lower ratio therefore does not indicate better physical scale preservation in this variant. The pairing comparison in the same table holds the images fixed and shuffles camera–interaction correspondences. Correct pairing lowers camera ratio and action error, but yields lower outcome gain. Appendix C gives the variant settings and additional controls.

4.3 Downstream Applications

Task success tests whether the representation improvements translate into useful policy targets. The two applications are π_0 (Black et al., 2024) on MetaWorld MT50 (McLean et al., 2026) and ABot-M0.5 (Chen et al., 2026) on RoboCasa365 (Nasiriany et al., 2026). Both applications use the frozen tokenizer through the policy interfaces in Section 3.4. Here, InterACT denotes the ALAM (Tang et al., 2026a) tokenizer adapted with our method. Appendix D provides implementation details.

Compared policies. MetaWorld baselines span diffusion policies (Diffusion Policy (Chi et al., 2025)), autoregressive policies (TinyVLA-H (Wen et al., 2025), RT-2 (Zitkovich et al., 2023), and RoboTron Mani (Yan et al., 2024)), video-augmented policies (GR-1 (Wu et al., 2023), PAD (Guo et al., 2024), and Evo-1 (Lin et al., 2025)), and flow-matching policies (SmolVLA (Shukor et al., 2025) and Evo-Depth (Lin et al., 2026)). RoboCasa baselines include GR00T-N1.5 (NVIDIA, 2026), Fast-WAM (Yuan et al., 2026), and Lingbot-VA (Li et al., 2026). Reported ALAM and ABot-M0.5 scores serve as references for the corresponding policies with our adapted tokenizer.

Main results. MetaWorld evaluates joint latent-and-action prediction across 50 manipulation tasks. Success is averaged within each difficulty tier, with equal weight across the four tiers. Table 5 reports

Table 5: **MetaWorld MT50 success (%).** Baseline tier scores and model sizes are reported values; Avg. is recomputed by equally weighting the four displayed tiers. DP: diffusion; AR: autoregressive; VA: video-augmented; FM: flow matching; LA: latent action. Bold marks the best value in each success column.

Method	Type	Size	Easy	Medium	Hard	Very hard	Avg.
Diffusion Policy	DP	0.3B	23.1	10.7	1.9	6.1	10.5
TinyVLA-H	AR	1.3B	77.6	21.5	11.4	15.8	31.6
RT-2	AR	7B	75.5	35.3	30.7	15.2	39.2
RoboTron Mani	AR	4B	85.5	67.7	76.7	81.0	77.7
GR-1	VA	0.2B	76.6	35.3	46.0	44.0	50.5
PAD	VA	–	81.8	65.1	56.7	**87.2**	72.7
Evo-1	VA	0.8B	89.2	76.8	77.2	79.2	80.6
π_0	FM	3B	71.8	48.2	41.7	30.0	47.9
SmolVLA	FM	2B	87.1	51.8	70.0	64.0	68.2
Evo-Depth	FM	0.9B	83.1	84.7	**87.3**	82.4	84.4
π_0 + ALAM	LA	3B	89.3	83.6	85.0	82.0	85.0
π_0 + ALAM + InterACT	LA	3B	**90.0**	**91.8**	86.7	86.0	**88.6**

Table 6: **RoboCasa365 atomic-seen success (%).** We evaluate the atomic-seen subset under the Target setting. Reference scores follow Chen et al. (2026). The last column uses the adapted ALAM tokenizer.

Subset	GR00T-N1.5	Fast-WAM	Lingbot-VA	ABot-M0.5 + ALAM	ABot-M0.5 + ALAM + InterACT
Atomic-seen	60.6	59.1	63.5	70.6	**76.11**

88.6% success for π_0 with InterACT targets, compared with the reported ALAM result of 85.0%. Success is higher in all four tiers, including medium tasks (83.6% to 91.8%) and very hard tasks (82.0% to 86.0%). RoboCasa evaluates the adapted tokenizer in ABot-M0.5 on atomic-seen kitchen tasks under the Target setting. Table 6 reports 76.11% success, compared with the reported 70.6% ABot-M0.5 reference, a gain of 5.51 percentage points. Together with MetaWorld, this result supports the use of the adapted latent targets in both joint latent–action and video–latent–action policies.

5 Conclusion

We presented InterACT, a framework for adapting pretrained latent actions through crossed camera–physics interventions. Matched transitions train a bounded residual to align views of the same interaction, while a frozen reference constrains physical distance scale. Separate gradient paths coordinate correction learning and encoder adaptation with physical preservation, interaction supervision, and continued video learning. The full adaptation achieves lower camera ratios and action errors and higher outcome gains than video continuation and ego-supervised adaptation across three controlled domains; controlled retrieval also improves over the pretrained tokenizer. The separate supervised comparison reveals a trade-off: the full method better preserves reference scale, but has lower outcome gains and domain-dependent action benefits. Together with the routing ablations, this result shows why camera ratios, physical scale, and predictive information should be assessed jointly. The frozen tokenizer also supplies policy targets on MetaWorld MT50 and RoboCasa365 atomic-seen tasks, with higher success than the reported latent-action references.

Limitations. InterACT learns alignment from sampled interventions, with physical retention defined relative to the frozen pretrained representation. Better scale retention does not ensure stronger outcome prediction, as the supervised comparison illustrates. The controlled data currently cover contact changes and translating cameras within a limited set of scenes. The improvements observed on controlled data do not extend to CALVIN retrieval or reconstruction on the six fixed video examples, so broader viewpoint and scene variation remains a challenge.

AI USE STATEMENT

Generative AI tools assisted with language polishing to improve clarity and readability. The authors take responsibility for the final text, claims, and reported results.

ETHICS STATEMENT

The authors have identified no additional ethical concerns requiring disclosure for this work.

REPRODUCIBILITY STATEMENT

The method and experimental protocols are described in Sections 3 and 4. Appendices A–D provide the training, evaluation, ablation, and downstream implementation details.

REFERENCES

Hongzhe Bi, Hengkai Tan, Shenghao Xie, Zeyuan Wang, Shuhe Huang, Haitian Liu, Ruowen Zhao, Yao Feng, Chendong Xiang, Yinze Rong, et al. Motus: A unified latent action world model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 35101–35113, 2026.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A Vision-Language-Action Flow Model for General Robot Control. *arXiv preprint arXiv:2410.24164*, 2024.

Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *Forty-first international conference on machine learning*, 2024.

Qingwen Bu, Yanting Yang, Jisong Cai, Shenyuan Gao, Guanghui Ren, Maoqing Yao, Ping Luo, and Hongyang Li. Univla: Learning to act anywhere with task-centric latent actions. *arXiv preprint arXiv:2505.06111*, 2025.

Xizhou Bu, Jiexi Lyu, Fulei Sun, Ruichen Yang, Zhiqiang Ma, and Wei Li. Laof: Robust latent action learning with optical flow constraints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 27334–27344, 2026.

Ronghan Chen, Yandan Yang, Zuojin Tang, Dongjie Huo, Tong Lin, Haoning Wu, Haoyun Liu, Yuzhi Chen, Lulu Zheng, Botai Yuan, et al. Abot-m0. 5: Unified mobility-and-manipulation world action model. *arXiv preprint arXiv:2607.00678*, 2026.

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

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

Shenyuan Gao, Siyuan Zhou, Yilun Du, Jun Zhang, and Chuang Gan. Adaworld: Learning adaptable world models with latent actions. *arXiv preprint arXiv:2503.18938*, 2025.

Yanjiang Guo, Yucheng Hu, Jianke Zhang, Yen-Jen Wang, Xiaoyu Chen, Chaochao Lu, and Jianyu Chen. Prediction with action: Visual policy learning via joint denoising process. In *Advances in Neural Information Processing Systems*, volume 37, pp. 112386–112410, 2024.

Youngjoon Jeong, Junha Chun, and Taesup Kim. Learning to act robustly with view-invariant latent actions. *arXiv preprint arXiv:2601.02994*, 2026.

Haoran Jiang, Jin Chen, Qingwen Bu, Li Chen, Modi Shi, Yanjie Zhang, Delong Li, Chuanzhe Suo, Hongyang Li, et al. Wholebodyvla: Towards unified latent vla for whole-body loco-manipulation control. In *International Conference on Learning Representations*, volume 2026, pp. 157438–157461, 2026a.

Yuxin Jiang, Yuchao Gu, Ivor W Tsang, and Mike Zheng Shou. Olaf-world: Orienting latent actions for video world modeling. *arXiv preprint arXiv:2602.10104*, 2026b.

Hanjung Kim, Lerrel Pinto, and Seon Joo Kim. Hierarchical latent action model. *arXiv preprint arXiv:2603.05815*, 2026.

Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Fei Han, Mingrui Yu, Zelin Gao, Nan Xue, Xing Zhu, et al. Causal world modeling for robot control. *arXiv preprint arXiv:2601.21998*, 2026.

Tao Lin, Yilei Zhong, Yuxin Du, Jingjing Zhang, Jiting Liu, Yinxinyu Chen, Encheng Gu, Ziyan Liu, Hongyi Cai, Yanwen Zou, et al. Evo-1: Lightweight vision-language-action model with preserved semantic alignment. *arXiv preprint arXiv:2511.04555*, 2025.

Tao Lin, Yuxin Du, Jiting Liu, Nuobei Zhu, Yunhe Li, Yuqian Fu, Yinxinyu Chen, Hongyi Cai, Zewei Ye, Bing Cheng, et al. Evo-depth: A lightweight depth-enhanced vision-language-action model. *arXiv preprint arXiv:2605.14950*, 2026.

Peidong Liu, Zhiyuan Xiang, Mingyang Li, Wenhao Li, Jiale Zhang, Jiahao Sun, and Jiawei Li. Wla3: World latent action modeling for semantics, dynamics, and kinematics. *arXiv preprint arXiv:2609.15870*, 2026.

Hao Luo, Ye Wang, Wanpeng Zhang, Haoqi Yuan, Yicheng Feng, Haiweng Xu, Sipeng Zheng, and Zongqing Lu. Joint-aligned latent action: Towards scalable vla pretraining in the wild, 2026. URL https://arxiv.org/abs/2602.21736.

Reginald McLean, Evangelos Chatzaroulas, Luc McCutcheon, Frank Röder, Tianhe Yu, Zhanpeng He, KR Zentner, Ryan Julian, Jordan Terry, Isaac Woungang, et al. Meta-world+: An improved, standardized, rl benchmark. *Advances in Neural Information Processing Systems*, 38, 2026.

Soroush Nasiriany, Sep Nasiriany, Abhiram Maddukuri, and Yuke Zhu. Robocasa365: A large-scale simulation framework for training and benchmarking generalist robots. In *International Conference on Learning Representations*, volume 2026, pp. 98643–98667, 2026.

NVIDIA. Gr00t n1.5: An improved open foundation model for generalist humanoid robots. *https://research. nvidia.com/labs/gear/gr00t-n1*$_5$/, 2026., 2026.

Zhenhao Shen, Jiaqi Liang, Jasper Lu, Feng Jiang, Yuran Wang, Chuanbo Wei, Jiayi Liu, Jianchun Yang, Qize Yu, Jiadi You, et al. Ld4wam: Learning latent dynamics from human videos for world action models. *arXiv preprint arXiv:2608.22403*, 2026.

Mustafa Shukor, Dana Aubakirova, Francesco Capuano, Pepijn Kooijmans, Steven Palma, Adil Zouitine, Michel Aractingi, Caroline Pascal, Martino Russi, Andres Marafioti, et al. Smolvla: A vision-language-action model for affordable and efficient robotics. *arXiv preprint arXiv:2506.01844*, 2025.

Zuojin Tang, Haoyun Liu, Xinyuan Chang, Changjie Wu, Dongjie Huo, Yandan Yang, Bin Liu, Zhejia Cai, Feng Xiong, Mu Xu, et al. Alam: Algebraically consistent latent action model for vision-language-action models. *arXiv preprint arXiv:2605.10819*, 2026a.

Zuojin Tang, Feifan Luo, Haoyun Liu, Botai Yuan, Dekang Qi, Ronghan Chen, Yandan Yang, Tong Lin, Xinyuan Chang, Mu Xu, et al. Dlam: Distributional latent actions with temporal constraints. *arXiv preprint arXiv:2607.27138*, 2026b.

Hangxing Wei, Xiaoyu Chen, Chuheng Zhang, Tim Pearce, Jianyu Chen, Alex Lamb, Li Zhao, and Jiang Bian. Learning additively compositional latent actions for embodied ai. *arXiv preprint arXiv:2604.03340*, 2026a.

Yufan Wei, Kun Zhou, Lingjun Mao, Zijun Zhang, Ziming Xu, Ziqiao Xi, Shuang Liang, Ruobing Han, Yuchen Yan, Xinyue Wang, et al. Causally debiased latent action model for embodied action conditioned world models. *arXiv preprint arXiv:2607.09185*, 2026b.

Junjie Wen, Yichen Zhu, Jinming Li, Minjie Zhu, Zhibin Tang, Kun Wu, Zhiyuan Xu, Ning Liu, Ran Cheng, Chaomin Shen, et al. Tinyvla: toward fast, data-efficient vision-language-action models for robotic manipulation. *IEEE Robotics and Automation Letters*, 10(4):3988–3995, 2025.

Hongtao Wu, Ya Jing, Chilam Cheang, Guangzeng Chen, Jiafeng Xu, Xinghang Li, Minghuan Liu, Hang Li, and Tao Kong. Unleashing large-scale video generative pre-training for visual robot manipulation. *arXiv preprint arXiv:2312.13139*, 2023.

Feng Yan, Fanfan Liu, Liming Zheng, Yufeng Zhong, Yiyang Huang, Zechao Guan, Chengjian Feng, and Lin Ma. Robotron-mani: All-in-one multimodal large model for robotic manipulation. *arXiv preprint arXiv:2412.07215*, 2024.

Seonghyeon Ye, Joel Jang, Byeongguk Jeon, Sejune Joo, Jianwei Yang, Baolin Peng, Ajay Mandlekar, Reuben Tan, Yu-Wei Chao, Bill Yuchen Lin, et al. Latent action pretraining from videos. *arXiv preprint arXiv:2410.11758*, 2024.

Tianyuan Yuan, Zibin Dong, Yicheng Liu, and Hang Zhao. Fast-wam: Do world action models need test-time future imagination? *arXiv preprint arXiv:2603.16666*, 2026.

Chuheng Zhang, Tim Pearce, Pushi Zhang, Kaixin Wang, Xiaoyu Chen, Wei Shen, Li Zhao, and Jiang Bian. What do latent action models actually learn? *arXiv preprint arXiv:2506.15691*, 2025.

Tianqiu Zhang, Muyang Lyu, Yufan Zhang, Fang Fang, and Si Wu. Dila: Disentangled latent action world models. *arXiv preprint arXiv:2605.15725*, 2026.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, et al. RT-2: Vision-language-action models transfer web knowledge to robotic control. In *Proceedings of the 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 2165–2183, 2023.

A Training Details

A.1 Model, Data, and Optimization

Both coverage settings use 3.9k updates, with $N_q = 640$ or $N_q = 5{,}120$. RGB sources use weights $1:1:1:3$; controlled quartets use a separate loader (Table 7). Table 8 lists the robot-source weights used in pretraining. During adaptation, the fixed robot-video pool is sampled as one source; these pretraining weights do not specify its internal composition.

Table 7: **Adaptation data and sampling.** Weights are normalized within the RGB loader; controlled quartets use a separate loader. Triplets contain three frames, and quartets contain four camera–physics transition pairs.

Data source	Fixed pool	Weight	Probability
EgoDex	25,000 triplets	1	$1/6$ (16.67%)
EgoVerse	25,000 triplets	1	$1/6$ (16.67%)
UMI GenRobot	25,000 triplets	1	$1/6$ (16.67%)
Robot videos (OXE + CALVIN)	100,000 triplets	3	$1/2$ (50%)
RGB total	**175,000 triplets**	**6**	**100%**
Controlled: all splits	1,024 quartets	—	—
Controlled: training ($N_q = 640$)	640 quartets	—	—

Table 8: **Robot-video sources in pretraining.** Shares normalize the source weights over all 11 datasets. Adaptation sampling is specified separately in Table 7.

Dataset	Weight	Share (%)
Open-X-Embodiment sources		
Fractal (RT-1)	150	31.71
BridgeData V2	50	10.57
Jaco Play	20	4.23
Berkeley Cable Routing	20	4.23
RoboTurk	10	2.11
TACO Play	5	1.06
NYU Door-Opening	5	1.06
Berkeley AutoLab UR5	5	1.06
TOTO	5	1.06
Viola	3	0.63
OXE subtotal	273	57.72
CALVIN	200	42.28
Total	**473**	**100.00**

Transitions within each quartet share the initial state and nominal action, with camera direction sampled independently of the action. Each initialization has eight direction–magnitude combinations, all assigned to one split (Table 9). Shuffling changes only pair assignments. Broader coverage varies object XY position by ± 0.06 m, yaw by ± 0.2 rad, and mass over 0.08–0.12 kg. Paired augmentation is applied to 200×200 video triplets; controlled transitions use the endpoints of 24-frame, 128×128 controlled clips. Both streams are resized to 224×224 RGB images; labels enter only the losses.

The frozen reference includes the encoder, transition tokens, and output map. The correction uses LayerNorm, a $128 \to 512$ layer, GELU, and a zero-initialized $512 \to 128$ layer. Three strided convolutions encode the initial image into 128 features. In Table 10, $\eta_{\max}, \eta_{\min}$ bound the learning rate; T denotes the selected update, and T_{sched} and T_{warm} the schedule and warmup lengths. We use AdamW with weight decay 10^{-4}, $\beta = (0.9, 0.99)$, epsilon 10^{-8}, gradient clipping at 1.0, bf16, and seed 42.

Table 9: **Intervention data splits.** Counts are quartets, with eight per initialization cluster and no cluster shared across splits. Probe fitting and regularization selection use only ID training and validation data.

Dataset	Train	Validation	Test	Total
$N_q = 640$	640	192	192	1,024
$N_q = 5{,}120$	5,120	1,536	1,536	8,192
Each evaluation domain	640	192	192	1,024

Table 10: **Adaptation settings.** Both intervention datasets share these settings. Batch sizes are effective counts of video triplets and controlled quartets.

Parameter	Value	Parameter	Value
(K, H)	$(7, 128)$	α	0.5
$\eta_{\max}, \eta_{\min}$	$10^{-5}, 10^{-6}$	τ	0.20
T, T_{sched}	3.9k, 7.8k	$[\rho_{\min}, \rho_{\max}]$	$[0.8, 1.2]$
T_{warm}	98	$[\nu_{\min}, \nu_{\max}]$	$[0.8, 1.2]$
B_{RGB}, B_q	1,024, 128	ϵ	10^{-6}
$\mu_{\text{pair}}, \mu_{\text{act}}$	$0.05, 0.10$	ϵ_{neg}	10^{-4}

Both settings use 16 NVIDIA H200 GPUs, with 16 triplets and two quartets per GPU and four accumulation steps. Video continuation uses 32 NVIDIA H200 GPUs with 32 triplets each and no accumulation, matching RGB sampling, effective batch, optimizer, and learning-rate schedule, but not FLOPs. Its two checkpoints share one run; evaluation at 3.9k updates retains the 7.8k-step schedule. Ego-supervised adaptation uses 3.9k updates with video objectives, egocentric motion supervision, RGB reference-distance regularization, and corrected-latent quantization. Joint backpropagation updates its encoder and residual.

A.2 Objective Implementation

Gradient paths follow Eq. (7). Loss weights default to one unless stated below.

Correction and physical preservation. The residual bound is enforced by rescaling the MLP output:

$$r_\phi(u) = \widetilde{r}_\phi(u) \min\left\{1,\ \frac{\alpha n_\epsilon(u)}{n_\epsilon(\widetilde{r}_\phi(u))}\right\}. \tag{A.1}$$

Distances use the smoothed RMS norm. Reference-direction matching uses cosine loss (weight 0.5); cross-view physical matching uses cosine loss and SmoothL1 on log norm ratios against detached reference-view differences (weight 0.5). For variance ratios, we use Eq. (4) with bounds $[\nu_{\min}, \nu_{\max}]$, a reference variance floor of ϵ^2, and weight 0.25. The residual guard penalizes the squared ratio of residual-difference and frozen physical-difference norms through z_{cam}.

Video learning. Reconstruction averages MSE over three forward triplet transitions; composition and reversal use raw-latent MSE. Straight-through quantization trains latents toward detached codes and codes toward detached latents, with weights 1 and 0.25 and gradient-based codebook updates. Corrected-latent quantization has weight 0.25. RGB distance matching uses SmoothL1 against frozen-reference distances (weight 0.1).

Interaction supervision. Ego supervision uses masked 14-coordinate bimanual translation, axis-angle, and gripper targets (weight 4). Source-specific affine calibration uses tanh-bounded scales 1 ± 0.25 and biases ± 0.25. Simulator actions encode direction, magnitude, intended contact, and unit duration; outcomes encode displacement, scalar-first quaternion as stored, and contact. SmoothL1 supervises these predictions, with predicted and target outcome displacement scaled by 10; action and outcome-difference targets remain unscaled. Outcome-difference prediction averages camera views and ranks correct targets against cyclic shifts. Binding replaces the latent with a detached one from

the preceding group at the same quartet position, keeping image and target fixed. Both rankings use MSE, margin μ_{pair}, and weight 0.25; binding averages scaled-coordinate errors over four transitions.

Cross-view action consistency. Head parameters and reference predictions are detached; SmoothL1 trains the alternate-view latent with path magnitude scaled by 10. Ranking uses the nearest distinct cross-device signature, concatenating both physical action descriptors, with margin μ_{act} and weight 0.5. Ranking and binding exclude mean squared signature separations below ϵ_{neg}; empty valid sets contribute zero.

B Representation Evaluation Details

Domains and probes. Table 11 defines the evaluation shifts. Probes use flattened continuous latents and the inputs specified in Section 4.1. We standardize features with ID reference-view training statistics (standard-deviation floor 10^{-6}), center targets, and fit separate float64 ridge probes for each model. Regularization is selected from $\{10^{-6}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 100\}$ on ID validation data; probes and transformations then remain fixed.

Table 11: **Controlled evaluation domains.** Cameras translate without rotating. "Same" denotes the ID setting; $\mathbf{c}_0$ and Δ_c denote initial position and translation span.

Setting	ID	Camera-OOD	Physics-OOD
Action directions	$\pm x,\ \pm y$	Same	XY diagonals
Path length (m)	0.10, 0.16	Same	0.12, 0.19
Mass (kg)	0.10	Same	0.16
Friction	$[0.45, 0.75]$	Same	$[0.30, 0.42]$
$\mathbf{c}_0$ (m)	$(1.05, -1.05, 0.86)$	$(0.82, -1.28, 1.02)$	Same
Δ_c (m)	0.40	0.52	Same

Metrics and uncertainty. Prediction RMSE uses coordinates scaled by their training RMS (floor 10^{-2}). Geometry uses unsmoothed RMS distances with denominator floor 10^{-8}. Paired 95% percentile intervals use 2000 cluster-bootstrap draws, retaining each cluster's eight quartets and sharing draws across models with probes fixed. We inspected Camera-OOD test scores during development; intervals cover test-cluster variation, not training-seed or model-selection uncertainty.

Retrieval and reconstruction. We rank 384 candidates from the reference camera for each of 384 queries from the alternate camera by latent RMS distance. Counterparts share the trajectory and initialization. Reconstruction uses 768 ID targets and each tokenizer's quantizer and decoder, with corrected latents for InterACT. PSNR averages image scores on $[0, 1]$ inputs; SSIM uses range one and AlexNet LPIPS uses $[-1, 1]$ RGB inputs. Figures 5 and 6 show six uniformly sampled queries and the first quartet from each of six evenly spaced clusters, without further filtering.

Video reconstruction uses two metadata-test examples each from Bridge, TACO Play, and Viola, at frame offsets 10, 25, and 35. CALVIN retrieval averages both static–gripper directions over 128 synchronized, non-overlapping ten-frame windows, sampled equally from four validation segments. Overlap with tokenizer pretraining has not been fully verified.

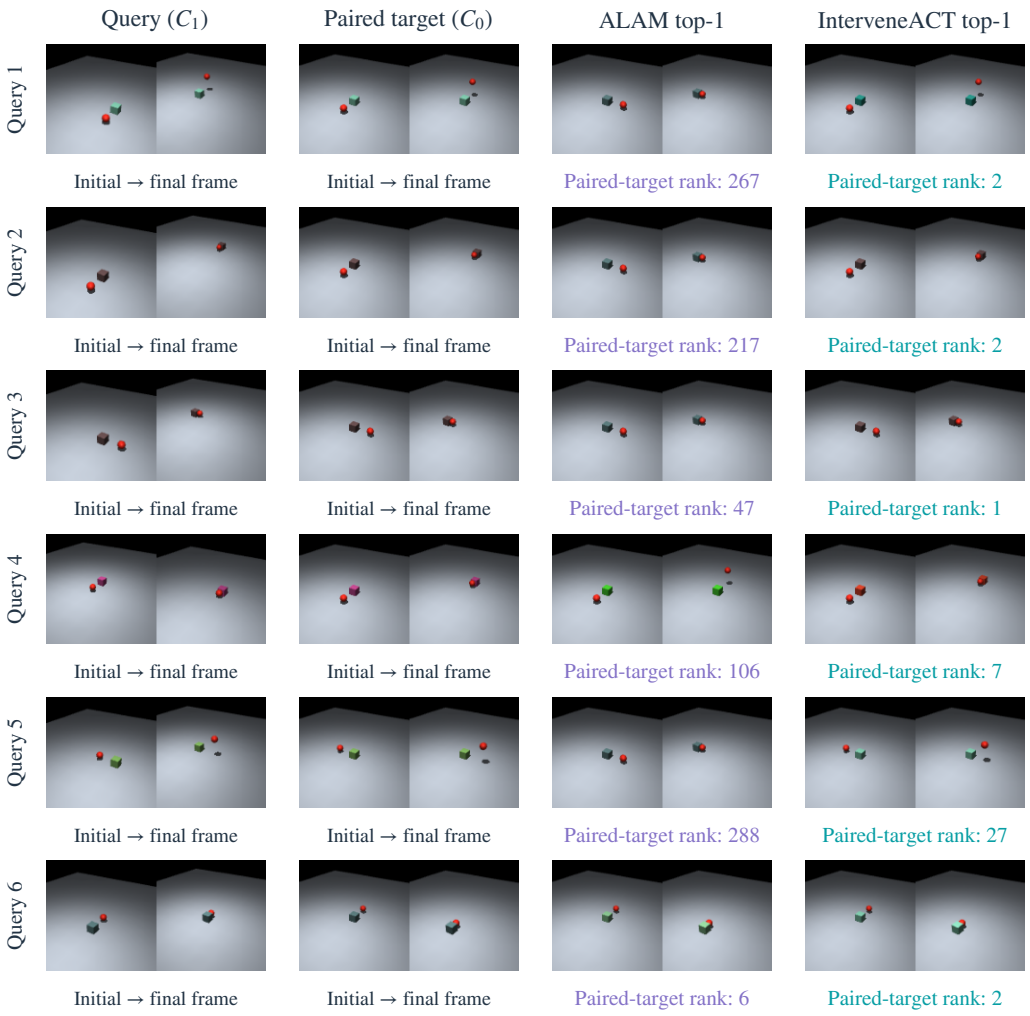


Figure 5: **Controlled cross-view retrieval.** Queries, paired transitions, and each model's nearest candidates are shown as initial–final image pairs. Ranks refer to the paired transition in the full gallery.

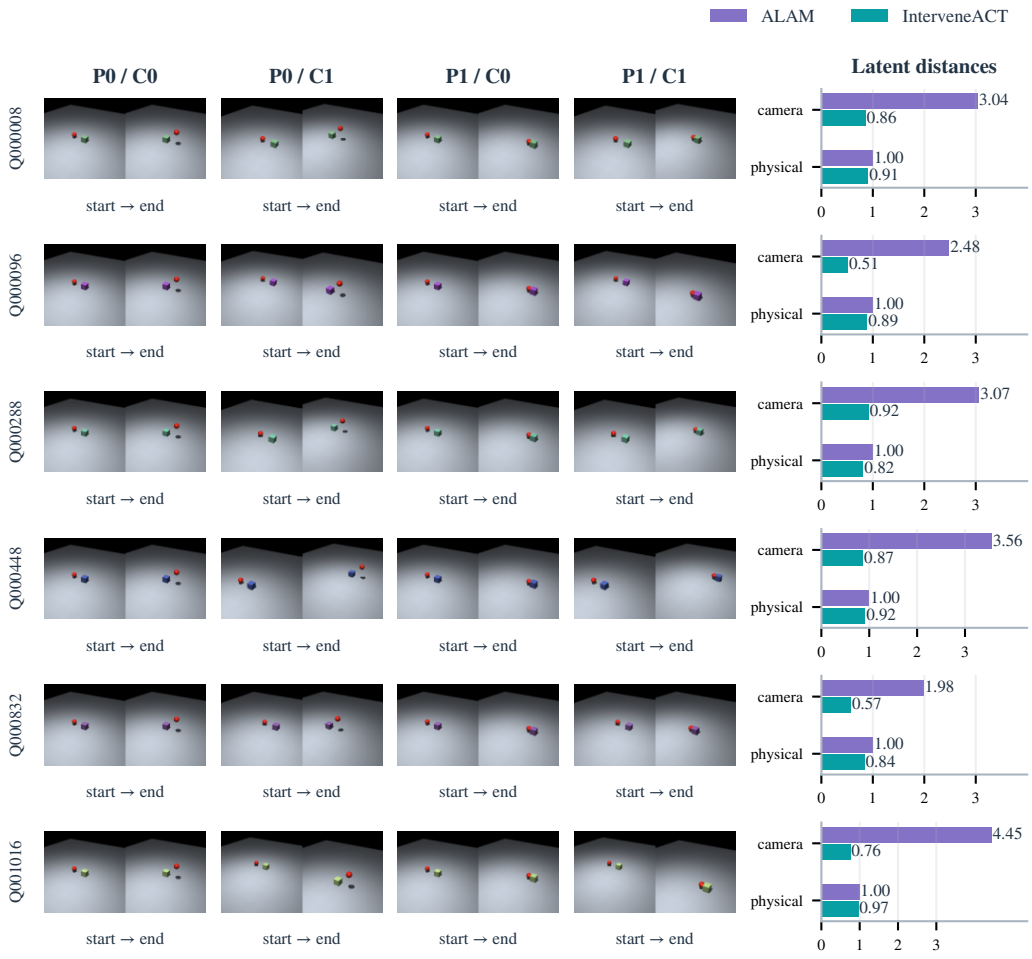


Figure 6: **Controlled quartet examples.** Each row shows two interactions under two cameras. Reference-normalized physical distances near one indicate scale retention; lower camera distances indicate closer alignment.

C Additional Representation Analyses

Increasing N_q from 640 to 5,120 with broader physical variation raises Physics-OOD outcome gain by 9.41 percentage points (paired 95% interval $[5.88, 13.07]$). The Camera-OOD change is -3.88 points ($[-9.48, 1.52]$), with an interval spanning zero. Figure 7 shows the training curves.

The full-model supervised comparison in the lower block of Table 1 is separate from the action-bottleneck studies below. The latter predict outcomes as a state baseline plus an action-conditioned residual; the main model uses the full corrected latent. Routing comparisons fix initialization, data, objectives, and 975 updates. For Table 12, students share the frozen prediction head of a correctly paired 975-step teacher and receive 975 further updates. Broader coverage improves camera ratio and outcome gain but raises action error; outcome consistency trades some of these gains for lower action error. "Full" denotes the variant's reference configuration.

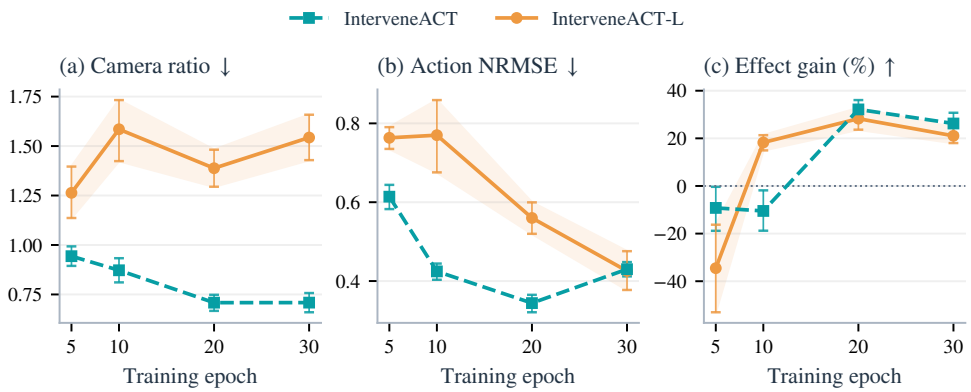


Figure 7: **Intervention coverage during training.** Camera-OOD metrics for $N_q = 640$ (squares) and 5,120 with broader physical variation (circles). Epochs 5, 10, 20, and 30 correspond to 975, 1.95k, 3.9k, and 5.85k updates. Shading shows 95% cluster-bootstrap intervals.

Table 12: **Coverage and outcome consistency.** Camera-OOD results for action-bottleneck students after 975 additional updates with a shared frozen teacher head.

InterACT	Coverage	$R_{\text{cp}} \downarrow$	$M \approx 1$	$E_a \downarrow$	G_e (%) $\uparrow$
w/o outcome consistency	Standard	0.95	0.95	0.53	−84.98
w/o outcome consistency	Broader	0.85	0.93	0.70	−2.55
Full	Broader	0.88	0.93	0.66	−5.35

D Downstream Evaluation Details

Both policies are trained on 32 AMD MI308 GPUs, using continuous targets from the frozen 3.9k-step tokenizer ($N_q = 640$). MetaWorld follows the joint π_0 interface in Section 3.4. For RoboCasa, we use a trainable $896 \to 30$ latent projection, three cameras, 16 state coordinates, and 12 action coordinates, with the geometry adapter disabled. Noise is added after projection. Scheduler-weighted MSE compares predicted velocities with detached noise-minus-target velocities, averaging valid coordinates within frames, then frames. Gradients through the noised input train the projection; loss weights and ordered sampling follow the original interface.