DREAMX: VIDEO DIFFUSION MODEL IS A UNIVERSAL LATENT ACTION LEARNER

Anonymous authors
Paper under double-blind review



Figure 1: **DreamX learns a shared action interface from human egocentric and robot videos.** Latent actions extracted from a pretrained video diffusion model capture motion across changes in embodiment and scene appearance. The same representation transfers source motion to a target video (left), decodes human motion into executable robot actions without paired human–robot training data (middle), and supports a compact latent world-action policy that predicts action codes and robot actions without generating future video (right).

ABSTRACT

Learning a universal action representation from video requires modeling dynamics that remain consistent across different executions of an action while being invariant to changes in embodiment and appearance. Existing latent action models (LAMs) infer actions from visual transitions, but their representations often conflate the underlying action with irrelevant visual variations, which limits transfer across embodiments. We introduce *DreamX*, a framework that transforms a pretrained video diffusion model into a universal latent action learner. Our approach extracts compact latent actions from the model's rich spatiotemporal features, compressing each latent video frame by 300× while using its generative prior to disentangle action-relevant dynamics from visual appearance. The resulting latent space serves as a shared action interface across human and robot embodiments, environments, and action parameterizations. On LARY-Bench, *DreamX* achieves state-of-the-art representation quality. More importantly, actions encoded in one setting remain effective in another: the universal latent space supports any-to-any video transfer across embodiments and environments, as well as zero-shot cross-embodiment action transfer. This shared interface also extends beyond representation learning to control, where a compact latent policy improves task performance while lowering the computational cost of video-diffusion-based control.

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.

1 INTRODUCTION

Generalist robot policies, including vision-language-action (VLA) models (Kim et al., 2024; Bjorck et al., 2025; Intelligence et al., 2025) and world action models (WAMs) (Kim et al., 2026; Ye et al., 2026), benefit from diverse robot demonstrations, but collecting such data at scale remains costly. Human egocentric videos offer a promising alternative as a scalable source of diverse manipulation behaviors, and recent works leverage these videos for policy learning (Li et al., 2025; Kareer et al., 2025). However, action supervision remains tied to the embodiment that produces it. Human videos lack robot action labels, and even robot demonstrations use different control spaces across embodiments. The same behavior can therefore correspond to different low-level actions, making action labels difficult to share directly across embodiments. Overcoming this mismatch calls for a common action representation that preserves shared motion and effects across embodiments.

Latent action models (LAMs) offer a promising route toward learning such a representation from video. Although humans and robots have different bodies and control spaces, their actions can all be observed as changes in a visual scene. LAMs typically use dedicated encoders to compress these changes into compact representations, called latent actions—a common representation of action across embodiments (Ye et al., 2025; Chen et al., 2025c; Bu et al., 2025b). These encoders are trained through future-frame reconstruction: a decoder predicts subsequent frames from the latent actions and an initial frame. However, future-frame reconstruction does not explicitly constrain latent actions to represent action alone. To reconstruct the full visual transition, existing LAMs may also encode domain-specific visual information associated with viewpoint, scene appearance, and embodiment. Capturing action-relevant structure shared across embodiments while separating it from domain-specific visual factors therefore remains a challenge.

Pretrained video diffusion models provide a promising foundation for addressing this challenge. First, through large-scale video generation, they learn rich spatiotemporal representations that capture motion, correspondence, geometry, and semantic structure (Nam et al., 2026; Wiedemer et al., 2025; Son et al., 2025). Such representations encode how scenes, objects, and agents evolve over time, making them a natural source of action-relevant dynamics. Second, these models possess a generative prior that enables future-frame synthesis conditioned on an initial frame and an additional signal. With scene appearance supplied by the initial frame, additional conditioning (e.g., language) focuses on steering the scene's evolution. Together, these priors suggest that action-relevant dynamics can be read out from the model's spatiotemporal representations and further shaped through future prediction, providing a basis for learning latent actions that transfer across embodiments.

We introduce *DreamX*, a framework for learning universal latent actions across embodiments by leveraging the priors of a pretrained video diffusion model. Our key idea is to use its spatiotemporal representations to encode action-relevant dynamics and its generative prior to supervise latent-action learning. To this end, we introduce learnable query tokens alongside video tokens and jointly encode them through the model's spatiotemporal attention. The queries read out dynamics across frames from the pretrained representations, condensing them into compact latent actions with $300\times$ fewer tokens than the video representation. We then use these latent actions, rather than text, to condition the same model to generate future frames from the initial frame. Given scene appearance from the initial frame and the pretrained generative prior, the latent actions are encouraged to encode the motion that leads to the observed future. Building on these latent actions, we develop a latent-action policy that predicts compact latent-action codes instead of future videos, reducing the computational burden of video-based policy models. Our experiments show that DreamX learns universal latent actions, enabling *any-to-any video transfer* and *human-to-robot action transfer* without paired training data. The latent-action policy further benefits from human video demonstrations, and DreamX achieves state-of-the-art performance among latent action models on LARYBench (Nie et al., 2026).

In summary, our contributions are as follows: (1) We introduce DreamX, a framework that leverages the spatiotemporal and generative priors of a pretrained video diffusion model to learn universal latent actions. (2) By leveraging the pretrained spatiotemporal prior, DreamX encodes motion-relevant structure into latent actions, achieving state-of-the-art performance among latent action models on LARYBench, with analyses revealing the motion information captured by the queries. (3) By exploiting priors of pretrained video diffusion models, DreamX learns universal latent actions that support *any-to-any video transfer* and *human-to-robot action transfer* without paired source–target data, as well as policy learning from human and robot videos.

2 Related Works

Latent action models. Latent action models (LAMs) learn action representations from visual transitions, enabling action-unlabeled videos to supervise downstream control. LAPO (Schmidt & Jiang, 2024) and Genie (Bruce et al., 2024) established the paradigm, recovering discrete latent actions through an inverse–forward dynamics bottleneck over adjacent frames. LAPA (Ye et al., 2025) further scales this recipe for vision-language-action pretraining, and Moto (Chen et al., 2025c) tokenizes visual motion for manipulation. Later work incorporates additional cues: UniVLA (Bu et al., 2025b) grounds latent actions with semantic visual features, villa-X (Chen et al., 2025b) integrates language and robot-state dynamics, and CoMo (Yang et al., 2026) and DiLA (Zhang et al., 2026) separate motion from visual content. Across these variants, the action space is learned from scratch on a local visual transition, often a single frame pair. Such supervision offers no view of how motion persists beyond one step, so the encoder entangles action with appearance and yields representations tied to their training embodiments.

Emergent capabilities of video generative models. Video generative models acquire spatiotemporal structure by modeling how scenes evolve over time. Their internal representations support correspondence, point tracking, optical flow, and geometric understanding without task-specific supervision (Nam et al., 2026; Vélez et al., 2025; Vedder et al., 2024; Huang et al., 2025), and text-conditioned models further exhibit semantic grounding over objects, behaviors, and interactions (Wiedemer et al., 2025). Rather than encoding physical laws explicitly, these representations capture motion, geometry, and semantics jointly: which entities are present, where they are, and how they move over time. This is precisely the composition a latent action must express, which is a structured change of a specific entity in a specific place, and it is acquired from web-scale video rather than from a single frame pair.

Video generative priors for latent action learning. Recent work connects these two lines. VPP (Hu et al., 2024) conditions a manipulation policy on the predictive visual representations inside a fine-tuned video diffusion model, but still learns the policy from action labels. AdaWorld (Gao et al., 2025) learns latent actions for adaptable video prediction, and VideoWorld2 (Ren et al., 2026b) pairs a pretrained video diffusion model for generation with a separate latent dynamics model. In both, the video model acts as a *generative* prior supplying pixels or rollouts, while the action representation is produced by an external module trained from scratch on frame-level transitions; so its spatiotemporal knowledge never shapes the action space itself. We instead read latent actions directly from the model's internal spatiotemporal representation, making maximal use of video pretraining and requiring no per-embodiment action encoder.

3 Method

Given a video and instruction, DreamX uses learnable queries to extract compact latent actions from a pretrained video diffusion backbone's spatiotemporal features. Future-frame reconstruction conditioned on these actions and the initial frame provides generative supervision (Sec. 3.2). The learned actions enable video and action transfer with a target initial frame or robot state (Sec. 3.3). For control, the DreamX policy predicts compact codes and low-level actions without generating future video (Sec. 3.4).

3.1 Preliminaries: Video Diffusion Models

We build on Cosmos-Predict2.5 (Ali et al., 2025), a pretrained video diffusion transformer. A causal video autoencoder compresses a clip into N latent frames, $\mathbf{V} = (V_0, \ldots, V_{N-1})$, where V_0 encodes only the initial frame. These frames are patchified into spatial tokens and jointly modeled through spatiotemporal attention. Through video pretraining, the model learns spatiotemporal priors that capture spatial context and temporal relationships across the clip. Conditioned on an initial frame and an additional signal such as text, collectively denoted by c, the model learns future-video generation through flow matching. Given clean video latents x and noise $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, we form $x_\tau = (1-\tau)x + \tau\epsilon$, with $\tau \in [0, 1]$, and minimize

$$\mathcal{L}_{\text{video}} = \mathbb{E}_{x,c,\epsilon,\tau}\left[\|v_\theta(x_\tau, \tau; c) - (\epsilon - x)\|_2^2\right], \tag{1}$$

where v_θ predicts the flow velocity.

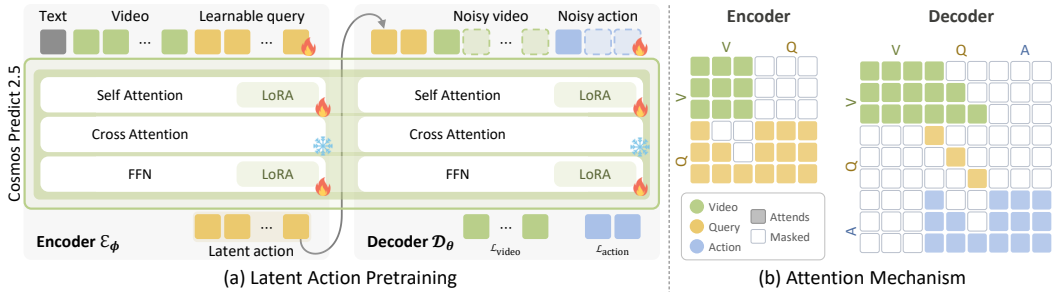


Figure 2: **DreamX architecture and attention masks.** (a) Learnable queries read out the pretrained video backbone's spatiotemporal representations to encode latent actions. A decoder initialized from the same pretrained backbone reconstructs future video from the latent actions and the initial frame, with auxiliary action supervision when labels are available. (b) Encoder and decoder attention masks control information flow among video, query, and action tokens. In the decoder, video and action tokens read the latent-action queries but do not attend to one another.

By optimizing this objective over large-scale video data, the model acquires a generative prior over scene appearance and plausible dynamics. When the initial frame supplies the scene's appearance, additional conditioning steers how the scene evolves. We exploit this property by using latent actions to steer future-frame reconstruction, whose objective supervises what the latent actions encode.

3.2 Learning Universal Latent Actions from Video Diffusion Priors

DreamX adapts a pretrained video diffusion backbone for two complementary roles: encoding observed dynamics into latent actions and supervising them through future-frame reconstruction to encourage separation of motion from visual appearance. Building on Sec. 3.1, we introduce learnable queries that read out the video representations, while the decoder reconstructs the future from these latent actions and the initial frame.

Latent Action Encoding from Spatiotemporal Dynamics. DreamX uses pretrained video model itself to encode latent actions. As shown in Figure 2(a), we append learnable query tokens $\mathbf{q}$ to the video token sequence and process them jointly through the backbone's spatiotemporal attention. We assign a group of K query tokens $\mathbf{q}_g$ to each latent frame V_g. Once contextualized by the video, their output features form the latent-action group Q_g. All N groups are extracted in a single forward,

$$Q = \mathcal{E}_\phi(\mathbf{q}; \mathbf{V}, \ell), \tag{2}$$

where ℓ is the language instruction supplied through the backbone's original cross-attention. As shown in Figure 2(b), query-to-video attention is frame-causal, so Q_g directly attends to $V_{0:g}$, while query-to-query attention is unrestricted. This lets the queries encode temporally grounded motion from the pretrained dynamics, with query-layout settings in Appendix A.2.

Generative Decoding for Action-Centric Latent Learning. Encoding from video alone may retain scene- and embodiment-specific visual cues. We therefore use the same pretrained video backbone as the encoder to supervise latent actions through future-frame reconstruction, conditioned on V_0 and $Q_{0:N-1}$ without text. As V_0 provides scene appearance and the pretrained generative prior captures plausible evolution, Q is encouraged to encode the motion needed to explain the future, rather than visual cues already available to the decoder.

As shown in Figure 2(a), the DreamX decoder generates future frames conditioned on $Q_{0:N-1}$, concatenated with the video tokens along the temporal sequence. When action labels are available, it also takes the proprioceptive state p_0 and decodes action chunks as an auxiliary task:

$$\hat{V}, \hat{A} = \mathcal{D}_\theta(Q_{0:N-1}, V_0, p_0). \tag{3}$$

Under the decoder attention mask in Figure 2(b), the queries act only as conditioning signals and do not attend to either the video or action tokens. The video and action streams do not attend to one another and share the latent actions as conditioning. Both streams attend to query prefixes: video frame V_t attends to $Q_{0:t}$, while action chunk A_m attends to $Q_{0:m}$. We train the decoder with the flow-matching objective of Eq. 1, applied to both video and action generation:

$$\mathcal{L} = \mathcal{L}_{\text{video}} + \lambda\, \mathcal{L}_{\text{action}}, \tag{4}$$

with $\lambda = 0$ for video clips without action labels.

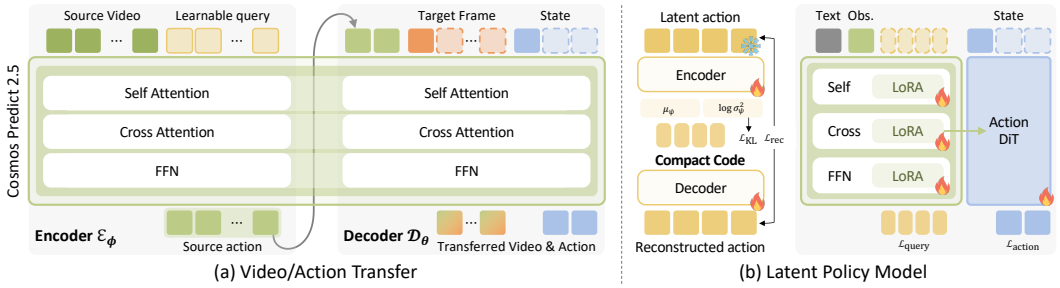


Figure 3: **Downstream applications of DreamX.** (a) **Motion transfer.** Source latent actions condition the DreamX decoder to generate video from a target initial frame or low-level actions from a target robot state, without paired source–target training data. (b) **DreamX policy.** A variational bottleneck compresses latent actions into compact codes while preserving the token layout. Conditioned on the current observation and instruction, the policy generates these codes alongside low-level actions through a separate DiT action head.

3.3 Motion Transfer via Latent Actions

Given a source video, we extract its latent actions using the frozen DreamX encoder, $Q^{\mathrm{src}} = \mathcal{E}_\phi(\mathbf{q}; \mathbf{V}^{\mathrm{src}}, \ell^{\mathrm{src}})$, which captures the motion to be transferred. As shown in Figure 3(a), we reuse the same Q^{src} in two complementary ways. For *video transfer*, we combine it with a target initial frame V_0^{tgt} and use the frozen DreamX decoder $\mathcal{D}_\theta$ to generate a target video that reproduces the source motion under the target scene and embodiment. For *action transfer*, we instead combine Q^{src} with the target robot state p_0^{tgt} and decode the same motion into executable low-level actions,

$$\hat{V}^{\mathrm{tgt}} = \mathcal{D}_\theta\big(Q^{\mathrm{src}}, V_0^{\mathrm{tgt}}\big), \qquad \hat{A}^{\mathrm{tgt}} = \mathcal{D}_\theta\big(Q^{\mathrm{src}}, p_0^{\mathrm{tgt}}\big). \tag{5}$$

Thus, the shared latent action realizes the same underlying motion either as a generated video in a new domain or as executable actions on a target robot, *without paired source–target datasets*.

3.4 DreamX Policy

For control, we repurpose the same pretrained video backbone used by DreamX to predict latent actions in place of future video. Although Q is compact in token count, each token retains the full backbone width, making direct diffusion high-dimensional (Chen et al., 2025a). We therefore compress Q into a low-dimensional code Z and train the DreamX policy to predict Z from the current observation and instruction.

Following (Gao et al., 2026), we learn a variational bottleneck to compress each token in Q from backbone width D to $d \ll D$, preserving the token layout, as shown in Figure 3(b):

$$Z = \mu_\psi(Q) + \sigma_\psi(Q) \odot \epsilon, \qquad \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{6}$$

where $\big(\mu_\psi(Q),\, \log \sigma_\psi^2(Q)\big) = f_\psi(Q)$ parameterizes the factorized Gaussian posterior $q_\psi(Z \mid Q)$ and $\hat{Q} = g_\psi(Z)$ reconstructs the original latent actions. Only f_ψ and g_ψ are trained with

$$\mathcal{L}_{\mathrm{FAE}} = \big\|\hat{Q} - Q\big\|_2^2 + \beta\, \mathrm{KL}\big(q_\psi(Z \mid Q) \,\|\, \mathcal{N}(\mathbf{0}, \mathbf{I})\big). \tag{7}$$

The reconstruction term encourages the codes to retain the information in Q, while the KL term regularizes their posterior toward a standard Gaussian.

With the learned bottleneck frozen, its compact codes serve as prediction targets for the DreamX policy. As shown in Figure 3(b), the policy generates these codes from the current frame V_0 and instruction ℓ, while a separate DiT action head predicts low-level actions conditioned on the generator's latent-action features, the same observation and instruction, and the state p_0:

$$\pi(Z, A \mid V_0, \ell, p_0). \tag{8}$$

Action tokens attend to latent-action features, but not vice versa. This separation lets the code generator learn from human video without robot action labels, while the action head learns from robot data. We train the policy with

$$\mathcal{L}_{\mathrm{policy}} = \mathcal{L}_{\mathrm{latent}} + \lambda\, \mathcal{L}_{\mathrm{action}}, \tag{9}$$

Table 1: **LARYBench action regression and classification.** Regression reports normalized action MSE and classification reports accuracy. **Bold** and underline denote the best and second-best results.

	Action regression ↓					Action classification ↑			
Method	CALVIN	VLABench	RoboCOIN	AgiBotWorld-Beta	Avg.	Atomic robot	Composite human	Composite robot	Avg.
Latent actions from 2 frames									
LAPA	0.96	0.95	0.96	1.00	0.97	22.27	14.61	23.64	20.17
UniVLA	0.82	0.74	0.94	0.97	0.87	18.62	19.08	18.56	18.75
villa-X	0.86	0.72	0.94	0.97	0.87	15.00	17.80	29.90	20.90
LAPA-MAGVIT2	0.59	0.36	0.80	0.83	0.65	33.12	59.70	29.53	40.78
LAPA-SigLIP2	0.57	0.30	0.86	0.88	0.65	46.83	54.74	29.44	43.67
LAPA-DINOv2	0.55	0.26	0.85	0.86	0.63	58.04	55.86	34.19	49.36
LAPA-DINOv3	0.50	0.25	0.82	0.84	0.60	56.27	64.19	27.04	49.17
Latent actions from video									
MoTo (Chen et al., 2025c)	0.43	0.13	0.54	0.68	0.44	41.57†	55.11	56.34	51.01
Adaworld (Gao et al., 2025)	0.64	0.26	0.94	0.94	0.69	31.58	49.8	51.48	44.29
VideoWorld2 (Ren et al., 2026a)	0.72	0.41	0.82	0.85	0.70	14.50	30.24	37.36	27.37
DiLA (Zhang et al., 2026)	0.43	0.09	0.51	0.52	0.38	58.51	65.47	62.76	62.25
Ours	**0.15**	**0.05**	**0.26**	**0.22**	**0.17**	**68.03**	**72.72**	**68.26**	**69.67**
Ours + FAE-32	0.23	0.07	0.31	0.26	0.22	65.90	69.49	66.06	67.15

where both terms follow Eq. 1, applied to compact codes Z and low-level actions A, respectively, with $\lambda = 0$ for human videos without action labels. Further details are provided in Appendix A.4.

4 Experimental Results

We evaluate whether DreamX learns universal latent actions by examining how well they capture motion (Sec. 4.1) and whether they can transfer across scenes and embodiments (Secs. 4.2–4.3). We further assess their utility for learning effective and efficient policies from human and robot demonstrations (Sec. 4.4), followed by ablations and analysis of the learned representation (Secs. 4.5–4.6).

Implementation details. We instantiate DreamX with Cosmos-Predict2.5 2B and pretrain on robot demonstrations from MolmoAct-BimanualYAM (Fang et al., 2026), Bridge (Walke et al., 2023), DROID (Khazatsky et al., 2024), and AgiBot (Bu et al., 2025a), together with human videos from EgoDex (Hoque et al., 2026) and EgoVerse (Punamiya et al., 2026). Videos are processed at 480×640 resolution with variable-length clips of up to 29 frames. We assign four query tokens to each VAE latent frame. We adapt the encoder and decoder using separate rank-16 LoRA modules for 78,000 training steps. Further training details are provided in Appendix A.2.

4.1 Representation Quality

We examine how well DreamX's latent actions capture motion on LARYBench (Nie et al., 2026). Regression asks whether the representation retains enough fine-grained motion to recover end-effector trajectories, and classification whether it carries enough semantic information to distinguish different motions. Table 1 compares DreamX with latent action models that encode either frame pairs or video clips. DreamX achieves the best performance across all benchmarks. Compared with the strongest baseline, DiLA, DreamX reduces regression MSE from 0.38 to 0.17 and increases classification accuracy from 62.25% to 69.67%. These results support leveraging video models as a source of action-relevant representations: reading latent actions directly from their pretrained priors yields representations that capture spatial, temporal, and semantic structure across embodiments. FAE-32, our 32-channel compressed representation (Sec. 3.4), also outperforms all baselines, indicating that even the compact codes retain substantial motion and semantic information. Further evaluation details are provided in Appendix B.4.

4.2 Video Transfer

We evaluate whether DreamX learns universal latent actions by transferring source motion across scenes and embodiments while preserving the appearance specified by a target initial frame. We examine any-to-any video transfer and quantitatively evaluate zero-shot human-to-robot transfer. Zero-shot evaluation details and additional transfer examples are provided in Appendices B.5 and D.1.

Any-to-any video transfer. Figure 4 shows motion transfer across environments for the same robot, between different robot embodiments, and from humans to robots, with a DAVIS example extending beyond manipulation. DreamX reproduces the source motion while preserving the target scene and embodiment. These results support the universality of the learned latent actions across embodiments and environments, extending to general-domain videos.

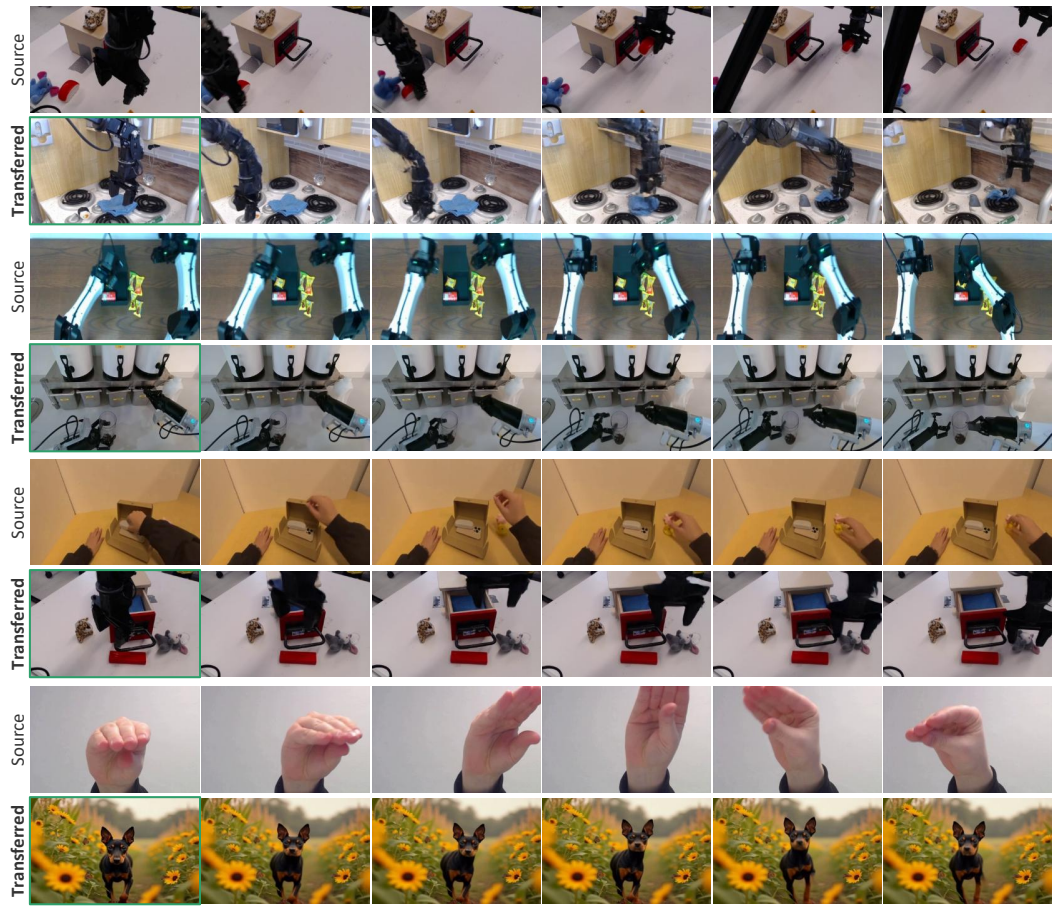


Figure 4: **Qualitative Any-to-Any video transfer.** Each example shows a source motion and the corresponding transferred video generated from a target-domain initial observation. These results demonstrate that the learned latent action transfers motion across substantial changes in embodiment and scene appearance.

Zero-shot human-to-robot transfer. We evaluate zero-shot human-to-robot transfer on HumanAndRobot (Xie et al., 2026), comparing with AdaWorld (Gao et al., 2025) and VideoWorld 2 (Ren et al., 2026b), both of which use diffusion-based video decoders. Paired robot recordings serve only as evaluation references; **no paired human–robot data is used for training**. DreamX outperforms both baselines on every metric in Table 2 and follows the source human motion while preserving the target robot and scene (Figure 5). In contrast, AdaWorld exhibits visual degradation in this out-of-distribution setting, while VideoWorld 2 occasionally introduces a human hand despite its appearance-disentangling design. These results suggest that DreamX better separates transferable motion from appearance, enabling more consistent human-to-robot transfer.

Table 2: **Zero-shot human-to-robot video transfer on H&R.** DreamX outperforms baselines across all metrics without paired human–robot training data

Method	FVD↓	LPIPS↓	PSNR↑	SSIM↑
AdaWorld	170.19	0.344	22.95	0.846
VideoWorld2	241.92	0.299	22.53	0.881
Ours	**57.44**	**0.179**	**26.52**	**0.917**

4.3 Action Transfer

We evaluate action transfer to examine whether our latent actions form a universal representation space that connects similar motions across embodiments. Following Sec. 3.3, we extract latent actions from a human video and decode them into low-level actions for the target robot. Execution is *open-loop without visual observations*: only the latent actions specify the motion, while proprioception provides the robot's initial state. Notably, this transfer is learned **without paired human–robot training data**, so the two embodiments are connected through the shared latent action space. Figure 6 shows that the robot reproduces the stroke structure of the letter "N," preserving the demon-

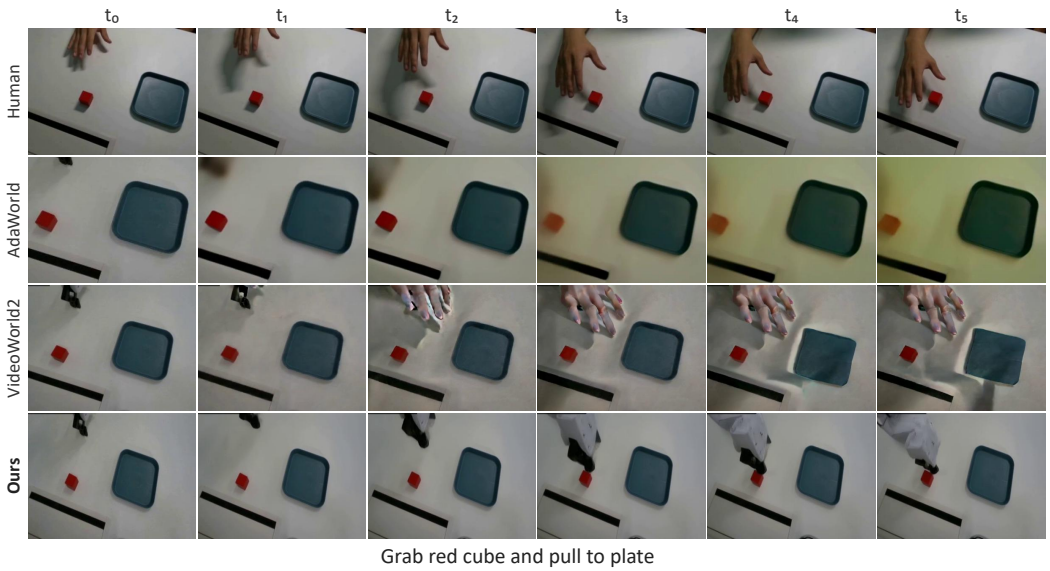


Figure 5: **Zero-shot human-to-robot video transfer on the HumanAndRobot benchmark.** DreamX preserves the source motion and target robot appearance, while the baselines exhibit visual degradation or human-hand leakage.

strated motion beyond a coarse task outcome. In single-arm pushing, it follows the forward-then-left sequence, retaining the direction change and temporal order. Bimanual pushing further shows corresponding movements of both arms, suggesting that the shared latent representation retains arm-specific motion and transfers it to the corresponding robot arms. Evaluation protocols and additional examples are provided in Appendices B.2 and D.2, respectively.

4.4 Policy Evaluation

We evaluate DreamX's effectiveness and efficiency in real-world control. Given the instruction *close the drawer*, the robot must close the upper drawer, the lower drawer, or both sequentially. We compare DreamX with Mimic-Video (Pai et al., 2025), both trained using human videos and robot demonstrations. During rollout, Mimic-Video generates future video, while DreamX generates compact latent actions alongside low-level actions.

Table 3: **Real-world drawer-closing SR (%) and inference latency.** DreamX achieves higher SR than Mimic-Video with $19.1\times$ faster inference, predicting compact latent actions instead of future video.

Method	Upper	Lower	Both	Avg.↑	Latency↓ (ms)
Mimic-Video	50.0	50.0	50.0	50.0	28,226
DreamX	90.0	100.0	100.0	96.7	1,474

Table 3 shows that DreamX achieves a subgoal success rate of 96.7%, compared with 50.0% for Mimic-Video. While Mimic-Video often reaches the drawer without completing the push, DreamX reliably completes both stages, including sequential drawer closure. DreamX also reduces inference latency from 28.23 s to 1.47 s per call—a $19.1\times$ speedup. These results suggest that action-centric representations learned from human and robot videos provide more effective conditioning for low-level control than future-video prediction. See Appendices B.3, C.1, and D.3 for evaluation details and additional results.

4.5 Ablation Study

We examine how auxiliary action supervision affects DreamX's latent actions, using the same subsets of MolmoAct, EgoVerse, and EgoDex to limit computational cost. Table 4 shows that action supervision slightly improves regression MSE ($0.24 \rightarrow 0.23$), while classification accuracy slightly decreases ($65.3\% \rightarrow 64.8\%$). The comparable performance suggests that DreamX captures motion and action semantics from video alone, without relying on action labels.

4.6 Analysis

Sensitivity to latent-action perturbations. We examine whether latent actions guide the motion reconstructed by the decoder. Keeping the initial frame and sampling noise fixed, we perturb the

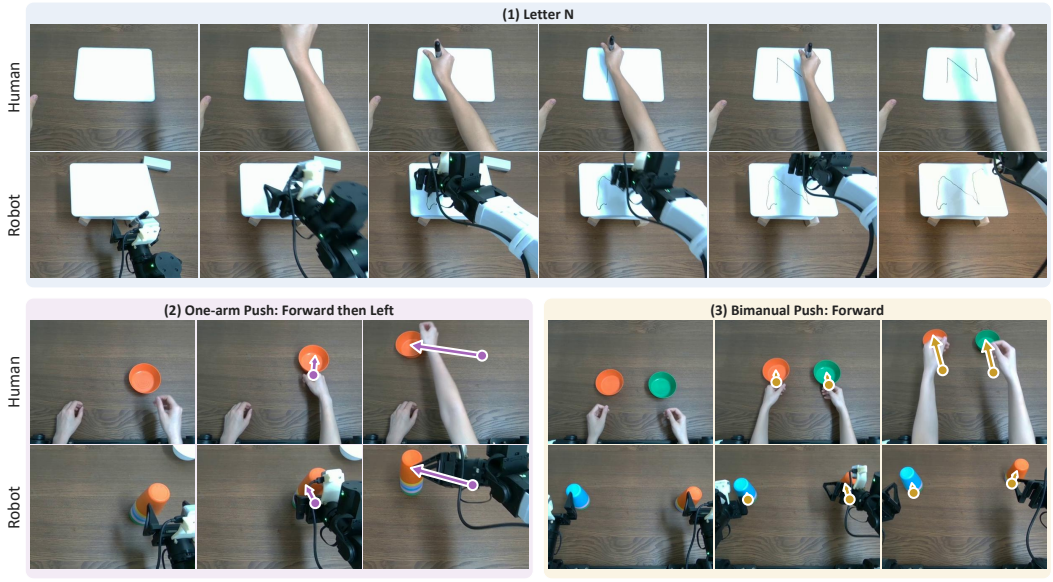


Figure 6: **Qualitative human-to-robot action transfer.** Given a human demonstration, DreamX extracts a latent action and decodes it into low-level robot actions without paired human–robot training data. We show diverse transferred behaviors spanning writing, single-arm pushing, and bimanual manipulation. Across substantial embodiment differences, the robot reproduces the motion structure of the human demonstration.

Supervision	Reg. avg. ↓	Cls. avg. ↑
RGB only	0.24	65.3
RGB + action	0.23	64.8

Table 4: **Encoder supervision ablation.** LARYBench regression MSE and classification accuracy.

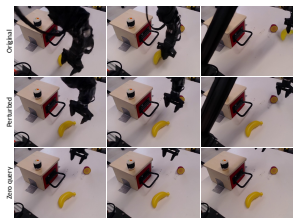


Figure 7: **Latent-action perturbation.**



Figure 8: **Query-to-video attention.**

latent actions or replace them with zeros. In Figure 7, the original latent actions reconstruct the grasping and lifting motion, whereas perturbation disrupts object manipulation and zeroing largely suppresses motion. The surrounding scene remains largely unchanged, supporting the role of latent actions in specifying motion beyond appearance. Quantitative results are provided in Appendix C.2.

What latent actions attend to. To examine where DreamX reads motion information, Figure 8 visualizes query-to-video attention during latent-action encoding of human and robot demonstrations. Attention highlights the hands and grippers involved in object interactions, suggesting that the queries read action-relevant regions across embodiments. Additional visualizations are provided in Appendix D.4.

5 Conclusion

We presented DreamX, a framework that leverages pretrained video diffusion priors to learn universal latent actions. Its spatiotemporal representations provide action-relevant dynamics, while its generative prior supervises latent-action learning through future-frame reconstruction. DreamX achieves state-of-the-art performance among latent action models on LARYBench and enables any-to-any video transfer and human-to-robot action transfer without paired training data. Predicting these compact latent actions also supports effective and efficient policy learning from human and robot demonstrations. Together, these results highlight pretrained video models as a foundation for learning reusable action representations across embodiments.

AI USE STATEMENT

We used generative AI tools to assist with manuscript drafting and editing for clarity and readability, code editing during implementation, and discussions during the research process. Research ideas, methodological choices, experimental designs, and interpretation of results were formulated, evaluated, and ultimately determined by the authors. All AI-assisted text and code were reviewed by the authors, and code changes were tested for correctness. We take responsibility for the final content of this work, including text, claims, code, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

All datasets used in this work are publicly available, including the human egocentric and robot demonstration datasets listed in Appendix A.1. Human videos may contain identifiable people or surroundings, so any redistributed examples should follow the source datasets' usage terms and privacy requirements. Video-transfer outputs are synthetic and should be identified as such. Cross-embodiment transfer and robot policies may fail outside the evaluated settings and require safety validation before deployment.

REPRODUCIBILITY STATEMENT

Section 3 describes the model, training objectives, transfer procedure, and policy. Appendix A documents the data sources and implementation details, while Appendix B specifies the benchmark, transfer, and real-world policy evaluation protocols. Supplementary videos illustrate video-transfer and action-transfer results; policy rollouts are provided in Appendix D.3.

REFERENCES

Arslan Ali, Junjie Bai, Maciej Bala, Yogesh Balaji, Aaron Blakeman, Tiffany Cai, Jiaxin Cao, Tianshi Cao, Elizabeth Cha, Yu-Wei Chao, et al. World simulation with video foundation models for physical ai. *arXiv preprint arXiv:2511.00062*, 2025.

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

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, Jisong Cai, Li Chen, Xiuqi Cui, Yan Ding, Siyuan Feng, Shenyuan Gao, Xindong He, Xuan Hu, Xu Huang, et al. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. *arXiv preprint arXiv:2503.06669*, 2025a.

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*, 2025b.

Junyu Chen, Dongyun Zou, Wenkun He, Junsong Chen, Enze Xie, Song Han, and Han Cai. Dc-ae 1.5: Accelerating diffusion model convergence with structured latent space. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1–10. IEEE, 2025a.

Xiaoyu Chen, Hangxing Wei, Pushi Zhang, Chuheng Zhang, Kaixin Wang, Yanjiang Guo, Rushuai Yang, Yucen Wang, Xinquan Xiao, Li Zhao, Jianyu Chen, and Jiang Bian. villa-x: Enhancing latent action modeling in vision-language-action models, 2025b. URL https://arxiv.org/abs/2507.23682.

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

Haoquan Fang, Jiafei Duan, Donovan Clay, Sam Wang, Shuo Liu, Weikai Huang, Xiang Fan, Wei-Chuan Tsai, Shirui Chen, Yi Ru Wang, et al. Molmoact2: Action reasoning models for real-world deployment. *arXiv preprint arXiv:2605.02881*, 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.

Yuan Gao, Chen Chen, and Jiatao Gu. One layer is enough: Adapting pretrained visual encoders for image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4688–4697, 2026.

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

Yucheng Hu, Yanjiang Guo, Pengchao Wang, Xiaoyu Chen, Yen-Jen Wang, Jianke Zhang, Koushil Sreenath, Chaochao Lu, and Jianyu Chen. Video prediction policy: A generalist robot policy with predictive visual representations. *arXiv preprint arXiv:2412.14803*, 2024.

Zixuan Huang, Xiang Li, Zhaoyang Lv, and James M. Rehg. How much 3d do video foundation models encode?, 2025. URL https://arxiv.org/abs/2512.19949.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. π0. 5: a vision-language-action model with open-world generalization, 2025. *URL https://arxiv. org/abs/2504.16054*, 1(2): 3, 2025.

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

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

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

Moo Jin Kim, Yihuai Gao, Tsung-Yi Lin, Yen-Chen Lin, Yunhao Ge, Grace Lam, Percy Liang, Shuran Song, Ming-Yu Liu, Chelsea Finn, et al. Cosmos policy: Fine-tuning video models for visuomotor control and planning. *arXiv preprint arXiv:2601.16163*, 2026.

Guangrun Li, Yaoxu Lyu, Zhuoyang Liu, Chengkai Hou, Jieyu Zhang, and Shanghang Zhang. H2r: A human-to-robot data augmentation for robot pre-training from videos. *arXiv preprint arXiv:2505.11920*, 2025.

Jisu Nam, Soowon Son, Dahyun Chung, Jiyoung Kim, Siyoon Jin, Junhwa Hur, and Seungryong Kim. Emergent temporal correspondences from video diffusion transformers. *Advances in Neural Information Processing Systems*, 38:123683–123727, 2026.

Dujun Nie, Fengjiao Chen, Qi Lv, Jun Kuang, Xiaoyu Li, Xuezhi Cao, and Xunliang Cai. Lary: A latent action representation yielding benchmark for generalizable vision-to-action alignment. *arXiv preprint arXiv:2604.11689*, 2026.

Jonas Pai, Liam Achenbach, Victoriano Montesinos, Benedek Forrai, Oier Mees, and Elvis Nava. mimic-video: Video-action models for generalizable robot control beyond vlas. *arXiv preprint arXiv:2512.15692*, 2025.

Ryan Punamiya, Simar Kareer, Zeyi Liu, Josh Citron, Ri-Zhao Qiu, Xiongyi Cai, Alexey Gavryushin, Jiaqi Chen, Davide Liconti, Lawrence Y Zhu, et al. Egoverse: An egocentric human dataset for robot learning from around the world. *arXiv preprint arXiv:2604.07607*, 2026.

Zhongwei Ren, Yunchao Wei, Xiao Yu, Guixun Luo, Yao Zhao, Bingyi Kang, Jiashi Feng, and Xiaojie Jin. Videoworld 2: Learning transferable knowledge from real-world videos, 2026a. URL https://arxiv.org/abs/2602.10102.

Zhongwei Ren, Yunchao Wei, Xiao Yu, Guixun Luo, Yao Zhao, Bingyi Kang, Jiashi Feng, and Xiaojie Jin. Videoworld 2: Learning transferable knowledge from real-world videos. *arXiv preprint arXiv:2602.10102*, 2026b.

Dominik Schmidt and Minqi Jiang. Learning to act without actions. In *International Conference on Learning Representations*, volume 2024, pp. 9379–9395, 2024.

Soowon Son, Honggyu An, Jisu Nam, Hyunah Ko, Chaehyun Kim, Dahyun Chung, Siyoon Jin, Jung Yi, Junhwa Hur, and Seungryong Kim. Probing and leveraging video diffusion transformer features for robust point tracking. *arXiv preprint arXiv:2512.20606*, 2025.

Kyle Vedder, Neehar Peri, Nathaniel Chodosh, Ishan Khatri, Eric Eaton, Dinesh Jayaraman, Yang Liu, Deva Ramanan, and James Hays. Zeroflow: Scalable scene flow via distillation, 2024. URL https://arxiv.org/abs/2305.10424.

Pedro Vélez, Luisa F. Polanía, Yi Yang, Chuhan Zhang, Rishabh Kabra, Anurag Arnab, and Mehdi S. M. Sajjadi. From image to video: An empirical study of diffusion representations, 2025. URL https://arxiv.org/abs/2502.07001.

Homer Rich Walke, Kevin Black, Tony Z Zhao, Quan Vuong, Chongyi Zheng, Philippe Hansen-Estruch, Andre Wang He, Vivek Myers, Moo Jin Kim, Max Du, et al. Bridgedata v2: A dataset for robot learning at scale. In *Conference on robot learning*, pp. 1723–1736. PMLR, 2023.

Thaddäus Wiedemer, Yuxuan Li, Paul Vicol, Shixiang Shane Gu, Nick Matarese, Kevin Swersky, Been Kim, Priyank Jaini, and Robert Geirhos. Video models are zero-shot learners and reasoners, 2025. URL https://arxiv.org/abs/2509.20328.

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

Jiange Yang, Yansong Shi, Haoyi Zhu, Mingyu Liu, Kaijing Ma, Yating Wang, Gangshan Wu, Tong He, and Limin Wang. Como: Learning continuous latent motion from internet videos for scalable robot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 42352–42363, 2026.

Seonghyeon Ye, Joel Jang, Byeongguk Jeon, Sejune Joo, Jianwei Yang, Baolin Peng, Ajay Mandlekar, Reuben Tan, Yu-Wei Chao, Bill Yuchen Lin, Lars Liden, Kimin Lee, Jianfeng Gao, Luke Zettlemoyer, Dieter Fox, and Minjoon Seo. Latent action pretraining from videos, 2025. URL https://arxiv.org/abs/2410.11758.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, et al. World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*, 2026.

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

This appendix provides implementation details, evaluation protocols, additional results and analysis.

A. IMPLEMENTATION DETAILS
- A.1 Training Data
- A.2 DreamX Pretraining
- A.3 Latent-Action Compression
- A.4 Policy Training and Inference

B. EXPERIMENTAL SETUP AND EVALUATION PROTOCOLS
- B.1 Real-World Robot Setup
- B.2 Action Transfer Details
- B.3 Policy Evaluation Details
- B.4 LARYBench Evaluation Details
- B.5 Zero-Shot Human-to-Robot Transfer Details

C. ADDITIONAL QUANTITATIVE RESULTS
- C.1 Policy Evaluation
- C.2 Representation Analysis

D. ADDITIONAL QUALITATIVE RESULTS AND ANALYSIS
- D.1 Any-to-Any Video Transfer
- D.2 Human-to-Robot Action Transfer
- D.3 Policy Rollouts
- D.4 Query-to-Video Attention

Video transfer, action transfer videos are provided in the supplementary materials.

A Implementation Details

A.1 Training Data

We pretrain DreamX on 2.19k hours of human and robot video spanning six datasets (Figure 9). Human demonstrations contribute 469 hours from EgoDex (Hoque et al., 2026) and EgoVerse (Punamiya et al., 2026), complementing robot demonstrations from Bridge (Walke et al., 2023), DROID (Khazatsky et al., 2024), AgiBot (Bu et al., 2025a), and MolmoAct-BimanualYAM (Fang et al., 2026).

A.2 DreamX Pretraining

We initialize DreamX from Cosmos-Predict2.5 2B and train at 480×640 resolution for 78,000 steps. Training clips contain 5–29 frames in increments of four, corresponding to 2–8 VAE latent frames. We assign four query tokens to each latent frame, yielding up to 32 query tokens per clip. The encoder and decoder use separate rank-16 LoRA modules, while the pretrained backbone weights remain frozen. Learning rates are 10^{-4} for the LoRA parameters and query embeddings and 5×10^{-5} for the remaining trainable parameters, with 500 warm-up steps. Video and action losses receive equal weights for action-labeled clips; clips without action labels use only the video loss.

A.3 Latent-Action Compression

We train a variational bottleneck on latent actions from the frozen DreamX encoder. A single self-attention block encodes each 2,048-dimensional token into a 32-dimensional code, and a six-layer transformer decoder reconstructs the original features. The token layout is preserved, mapping a $32 \times 2{,}048$ representation to 32×32 compact codes. Training uses feature-reconstruction MSE in a fixed normalized feature space and a KL regularizer with weight $\beta = 0.01$. We train for 50,000 steps with a learning rate of 10^{-4} and 500 warm-up steps. After training, we use the posterior mean to obtain deterministic compact codes for downstream learning and evaluation.

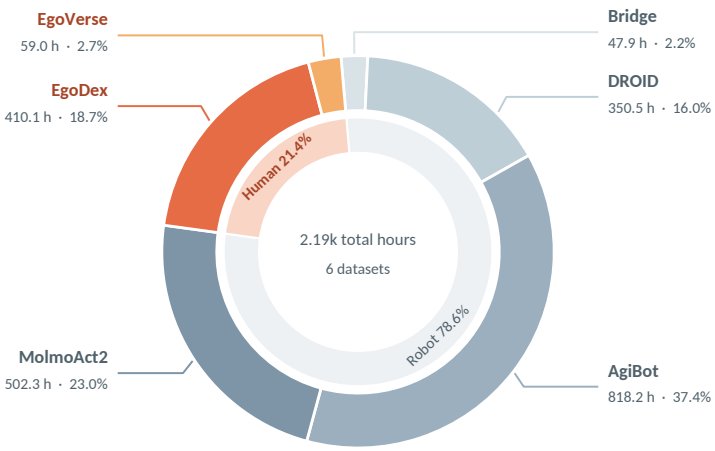


Figure 9: **Pretraining data composition.** DreamX learns from six datasets spanning human and robot demonstrations. Sector areas represent video duration, with human datasets highlighted in warm colors.

A.4 Policy Training and Inference

To convert the pretrained video model into a latent-action generator, we train it to predict compact latent-action codes conditioned on the current observation and language instruction. We use approximately one-third of the DreamX pretraining data and optimize rank-128 LoRA modules in self-attention, cross-attention, and feed-forward layers, together with trainable input and output projections. The generator is trained for 100,000 steps on two NVIDIA H200 GPUs with a global batch size of 60, a learning rate of 10^{-4}, and 1,000 warm-up steps. The Action DiT is not separately pretrained; it is introduced during task-specific policy learning and trained jointly with the generator. Task-specific training settings are provided in Appendix B.3.

B Experimental Setup and Evaluation Protocols

B.1 Real-World Robot Setup

We conduct both action-transfer and policy experiments on a bimanual YAM robot. The action space consists of 14-dimensional absolute commands, comprising six joint positions and one gripper command per arm. We leverage one Intel RealSense D405 camera for observation, which is mounted on top view.

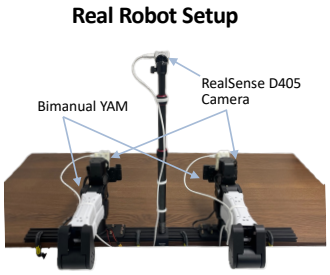


Figure 10: **BimanualYAM setup.**

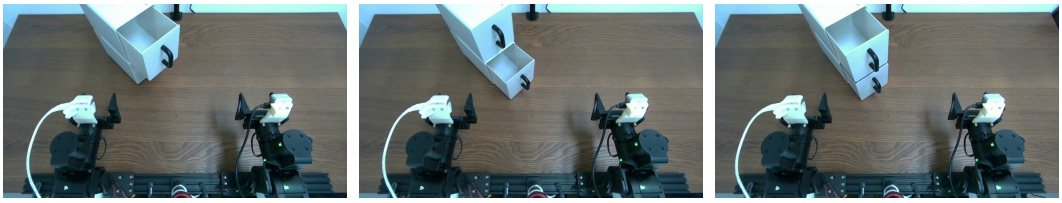
B.2 Action Transfer Details

We extract latent actions from human demonstrations using the frozen DreamX encoder and decode them into robot commands. The resulting trajectory is executed *open-loop, without visual feedback*, conditioned on the human latent actions and the robot's initial proprioceptive state.

We evaluate three types of motion transfer. **Writing** tests whether latent actions preserve fine-grained spatial trajectories and stroke order; a marker attached to the right gripper reproduces the demonstrated shape. **Single-arm pushing** tests whether the representation captures changes in motion direction and their temporal order, requiring the robot to follow a forward-then-left sequence. **Bimanual manipulation** tests whether latent actions retain arm-specific motion and transfer it to the corresponding robot arms. Additional examples are provided in Section D.2.

B.3 Policy Evaluation Details

Task and observations. We evaluate three drawer-closing configurations: only the upper drawer open, only the lower drawer open, and both drawers open, as illustrated in Figure 11. All configurations use the instruction *close the drawer*. When both drawers are open, the robot must close the upper drawer before the lower drawer. A fixed top-view RGB camera observes the workspace.



(a) Upper drawer open (b) Lower drawer open (c) Both drawers open

Figure 11: **Real-world drawer-closing setup.** Initial observations for the three task configurations, all conditioned on *close the drawer*. When both drawers are open, the robot must close the upper drawer before the lower drawer.

Training. We compare DreamX with Mimic-Video (Pai et al., 2025), using human videos and robot demonstrations for task-specific learning. DreamX starts from the pretrained latent-action generator, while Mimic-Video starts from a video backbone pretrained on MolmoAct. Each method then incorporates an action head for task-specific learning, and both are trained for 30,000 steps. For DreamX, human videos supervise latent-action prediction, while robot demonstrations additionally supervise low-level actions. During rollout, DreamX generates compact latent actions alongside low-level actions, whereas Mimic-Video generates future video.

Evaluation. We evaluate reaching and pushing as two subgoals: reaching measures whether the robot approaches the drawer for interaction, and pushing measures whether it completes the required closure. The reported subgoal success rate averages the two completion rates within each configuration, and the overall average weights the three configurations equally. We additionally report inference latency per policy call. Separate reaching and pushing results are provided in Section C.1.

B.4 LARYBench Evaluation Details

Evaluation protocol. We freeze all representation models and train a common attentive probe for action regression and classification. Each input token is projected to 768 dimensions through a linear layer, layer normalization, and GELU. For representations spanning multiple transitions, learned temporal embeddings identify the transition positions before the temporal and token dimensions are flattened. A learned-query attentive pooler aggregates the tokens into a single feature for the regression or classification head.

Feature extraction. The regression benchmark originally provides only two frames per sample. Using the accompanying metadata, we recover the corresponding video sequences from the source datasets, including the intervening frames, for video-based models.

We extract each model's native representation without fine-tuning and preserve its temporal and token structure. Pair-based models encode consecutive frame pairs, while video-based models encode clips using their supported temporal windows. Features are precomputed for probe training. Table 5 summarizes the input frames, extracted features, and temporal aggregation for each method.

B.5 Zero-Shot Human-to-Robot Transfer Details

We evaluate on 1,008 clips spanning 21 tasks from HumanAndRobot (Xie et al., 2026), selecting 48 clips per task. Each method receives a human source video and the initial frame of its paired robot recording. The remaining robot frames serve only as evaluation references; *no paired human–robot data is used for training*. We compare DreamX with AdaWorld (Gao et al., 2025) and VideoWorld 2 (Ren et al., 2026b) using their native generation horizons.

Table 5: **Feature extraction protocol for LARYBench.** All representation models are frozen and only the downstream probe is trained.

Method	Native temporal unit	LARYBench input	Feature extraction	Temporal aggregation
LAPA	Adjacent frame pair	9 frames → 8 pairs	4 latent tokens per pair	Temporal attentive pooling
UniVLA	Frame pair	9 frames → 8 pairs	DINO-LAM latent tokens	Temporal attentive pooling
villa-X	Video clip	9-frame clip	IDM latent representation	Attentive pooling
LAPA-MAGVIT2	Frame pair	9 frames → 8 pairs	16 visual tokens per pair	Temporal attentive pooling
LAPA-SigLIP2	Frame pair	9 frames → 8 pairs	16 visual tokens per pair	Temporal attentive pooling
LAPA-DINOv2	Frame pair	9 frames → 8 pairs	16 visual tokens per pair	Temporal attentive pooling
LAPA-DINOv3	Frame pair	9 frames → 8 pairs	16 visual tokens per pair	Temporal attentive pooling
MoTo	Adjacent frame pair	9 frames → 8 pairs	Quantized motion tokens	Temporal attentive pooling
AdaWorld	Adjacent frame pair	9 frames → 8 pairs	Latent action per pair	Temporal attentive pooling
VideoWorld2	Causal video transition	9-frame clip	FSQ latent action codes	Attentive pooling
DiLA	Fixed video window	16-frame clip	15 latent transitions	Attentive pooling
Ours	Variable-length video clip	Selected video clip	Latent-action query tokens	Attentive pooling

For a common evaluation horizon, we retain the first 20 frames of each output and exclude the conditioning frame, yielding 19 generated frames per clip. We report FVD using a Kinetics-400-pretrained I3D network, VGG-based LPIPS, PSNR, and SSIM against the corresponding robot reference videos.

C Additional Quantitative Results

C.1 Policy Evaluation

Subgoal Completion Table 6 reports reaching and pushing completion separately. Averaging each pair gives the per-configuration SR in Table 3: 50.0% for Mimic-Video in all configurations, and 90.0%, 100.0%, and 100.0% for DreamX.

Table 6: **Reaching and pushing completion rates (%) for real-world drawer closing.** Each subgoal contributes equally to the SR reported in the main table.

	Upper		Lower		Both	
Method	Reach	Push	Reach	Push	Reach	Push
Mimic-Video	80	20	100	0	100	0
DreamX	100	80	100	100	100	100

Policy Learning from Human Demonstrations We additionally train a policy using *only human videos for the upper-only drawer-closing configuration*, while retaining robot demonstrations for the lower-only and both-open configurations. Human videos supervise latent-action prediction without low-level action labels. As shown in Table 7, the policy achieves 100% success in upper drawer task even if trained only with human dataset.

Table 7: **Policy learning with human-only demonstrations of the upper-only configuration.** Task columns report approach/push completion rates (%); SR denotes full-task success (%).

Method	Upper	Lower	Both	SR↑
DreamX	100/100	100/100	100/100	100

C.2 Representation Analysis

Motion Structure across Embodiments We examine whether latent actions exhibit common directional structure across robot embodiments using 2,362 RoboCOIN clips spanning ten robot types and six end-effector translation directions.
Figure 12 visualizes 60 direction-conditioned latent-action centroids using t-SNE. Centroids for the same direction exhibit similar organization across robot types. This aggregate analysis supports shared directional structure across embodiments.

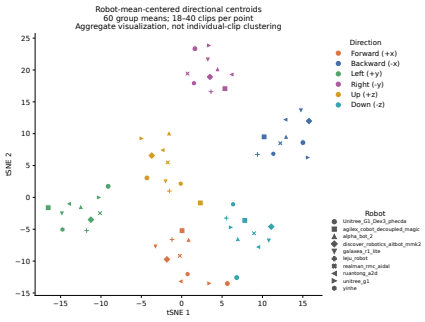


Figure 12: **Directional structure across robot embodiments.** Colors denote motion directions and marker shapes denote robot types.

Effect of Latent-Action Conditioning We quantify how changes to latent actions affect decoded motion and appearance. For latent actions $Q \in \mathbb{R}^{N_q \times D}$, we perturb the decoder's conditioning as

$$\widetilde{Q}_{i,d} = Q_{i,d} + \alpha s_d \epsilon_{i,d}, \qquad \epsilon_{i,d} \sim \mathcal{N}(0,1), \tag{10}$$

where s_d is the channel-wise standard deviation estimated from training features. We evaluate $\alpha \in \{0, 0.1, 0.3, 1.0\}$ and a zero-query control on eight selected high-motion clips, with two decoder seeds per clip. Within each comparison, we fix the initial frame, decoder noise, denoising schedule, and perturbation draw, rescaling only the perturbation magnitude. The stronger qualitative perturbation in Figure 7 uses $\alpha = 2$.

Table 8 reports mean changes relative to the unperturbed reconstruction, and Figure 13 shows their distributions across clips and decoder seeds. We compute Farnebäck optical flow at 160×120 and measure mean flow-vector deviation and motion magnitude relative to the unperturbed output. RGB differences use intensities in $[0, 1]$ and exclude the conditioning frame. A fixed static-region mask selects pixels below the 40th percentile of temporal RGB variation in the reference video, followed by 5×5 erosion.

Increasing perturbation strength produces larger changes in decoded motion, while static-region appearance changes remain smaller than whole-image changes. Zeroing the latent actions reduces the mean per-sample motion ratio to 0.456. Together with the qualitative examples, these measurements support the role of latent actions in steering motion while the decoder retains much of the scene appearance.

Conditioning	Motion ratio	Flow deviation	RGB L1	Static RGB L1
Original	1.000	0.000	0.0000	0.0000
Noise ($\alpha = 0.1$)	1.000	0.141	0.0054	0.0030
Noise ($\alpha = 0.3$)	0.995	0.211	0.0077	0.0034
Noise ($\alpha = 1.0$)	0.946	0.481	0.0215	0.0077
Zero queries	0.456	0.866	0.0758	0.0227

Table 8: **Sensitivity to latent-action perturbations.** Means over eight clips and two decoder seeds. Flow deviation (pixels/frame) and RGB L1 are measured against unperturbed reconstructions. Motion ratio is averaged per clip and seed; no perturbation has ratio 1.

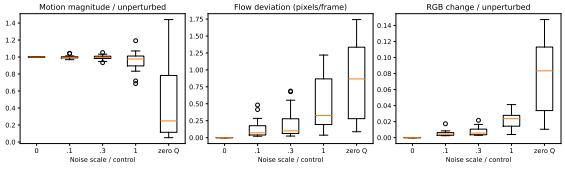


Figure 13: **Distribution of perturbation effects.** Motion ratio, flow deviation, and RGB change for the same eight clips and two decoder seeds as Table 8. Boxes show quartiles and medians; whiskers extend to observations within 1.5 times the interquartile range, with outliers shown individually.

D Additional Qualitative Results and Analysis

D.1 Any-to-Any Video Transfer

D.1.1 In-domain Transfer

Figure 14 provides additional any-to-any video transfer examples across embodiments and environments, complementing Figure 4 in the main paper.

D.1.2 Additional Zero-Shot Human-to-Robot Transfer

Figure 15 provides additional zero-shot human-to-robot video transfer comparisons on HumanAndRobot, complementing Figure 5. All methods use the evaluation protocol in Section B.5.

D.2 Human-to-Robot Action Transfer

Figure 16 provides additional human-to-robot action transfer examples for writing and single-arm manipulation, complementing Figure 6. The experimental setup is described in Section B.2.

D.3 Policy Rollouts

Figure 17 provides illustrative real-world drawer-closing rollouts for the upper-only, lower-only, and both-open configurations.

D.4 Query-to-Video Attention

Figure 18 provides additional query-to-video attention visualizations across human and robot demonstration sequences, complementing Figure 8. We average query-to-video attention over all heads, the four query slots within each temporal group, and the final seven transformer blocks.

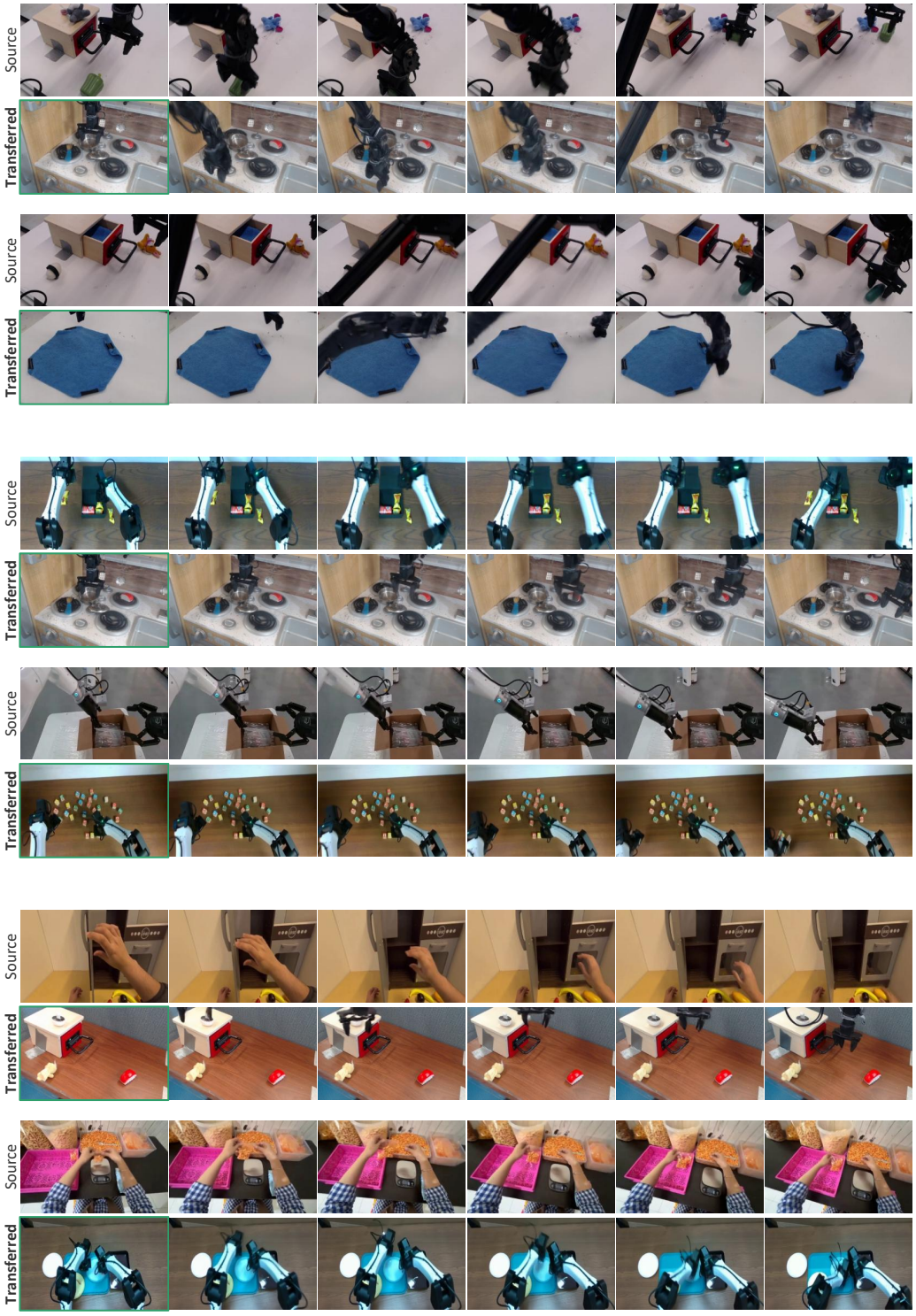


Figure 14: **Additional qualitative Any-to-Any video transfer results.** Each example shows a source motion and the corresponding transferred sequence generated from a target-domain initial frame.

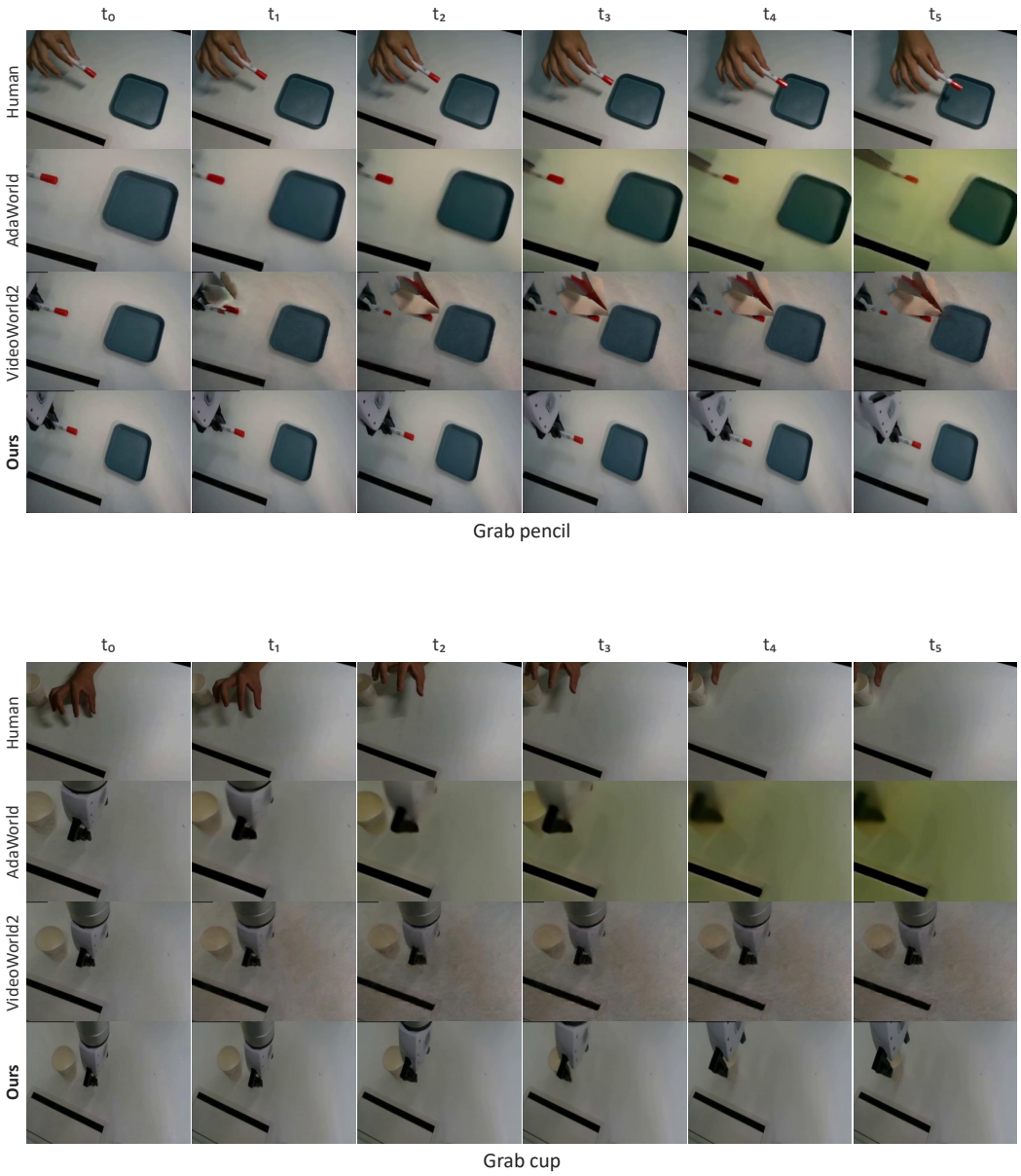


Figure 15: **Additional zero-shot human-to-robot video transfer results on the HumanAndRobot benchmark.**

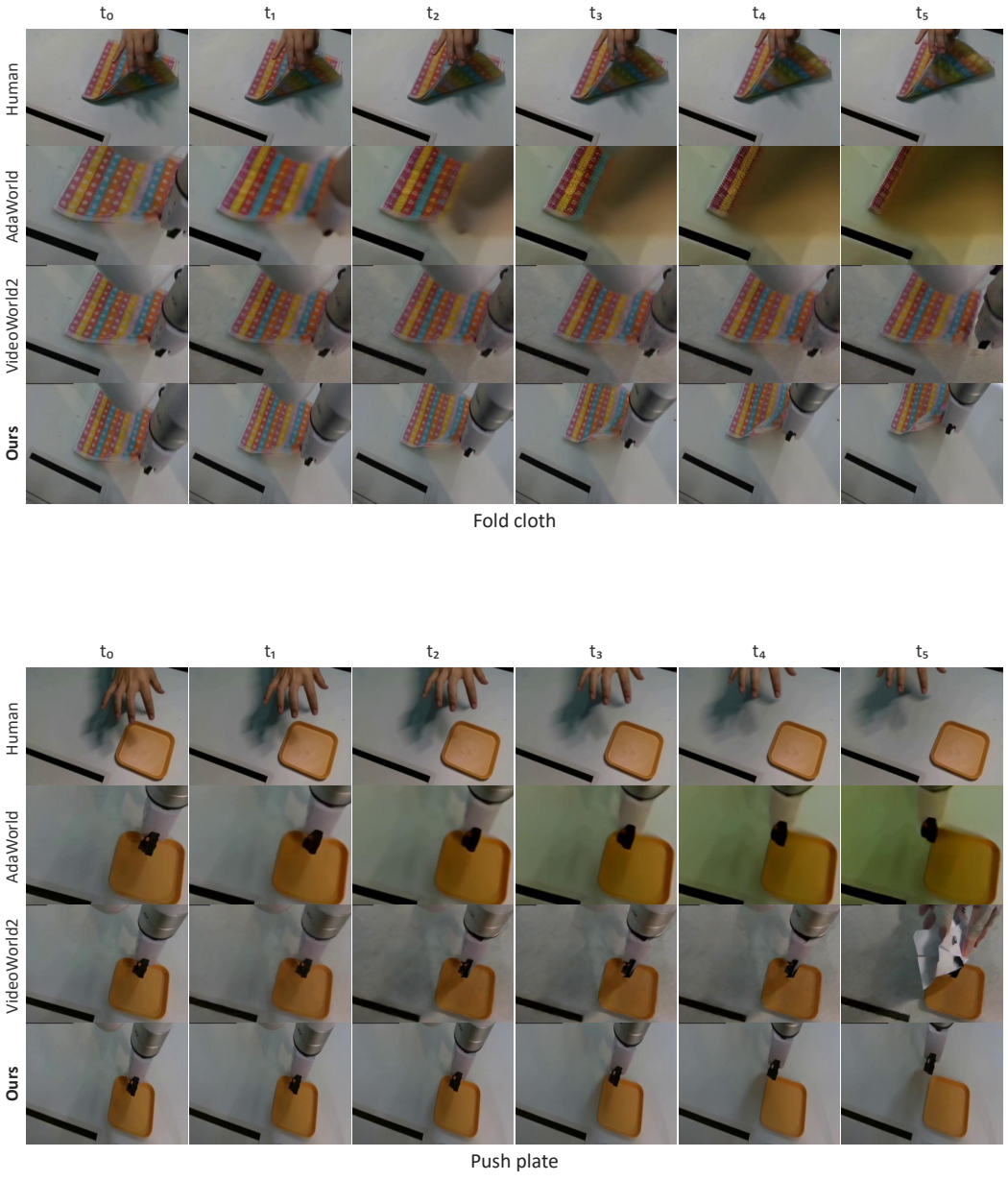


Figure 15: **Additional zero-shot human-to-robot video transfer results on the HumanAndRobot benchmark (continued).**

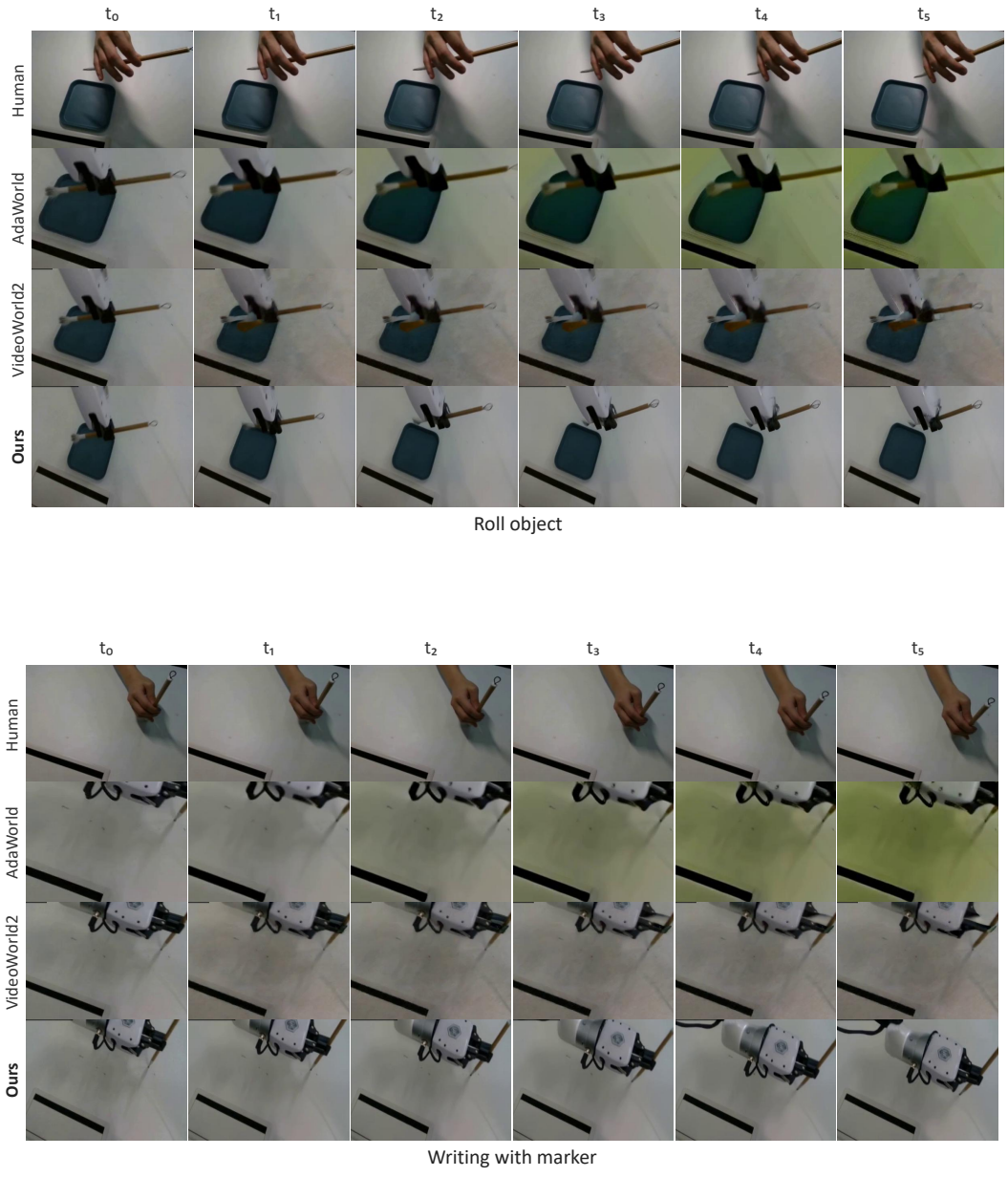


Figure 15: **Additional zero-shot human-to-robot video transfer results on the HumanAndRobot benchmark (continued).**

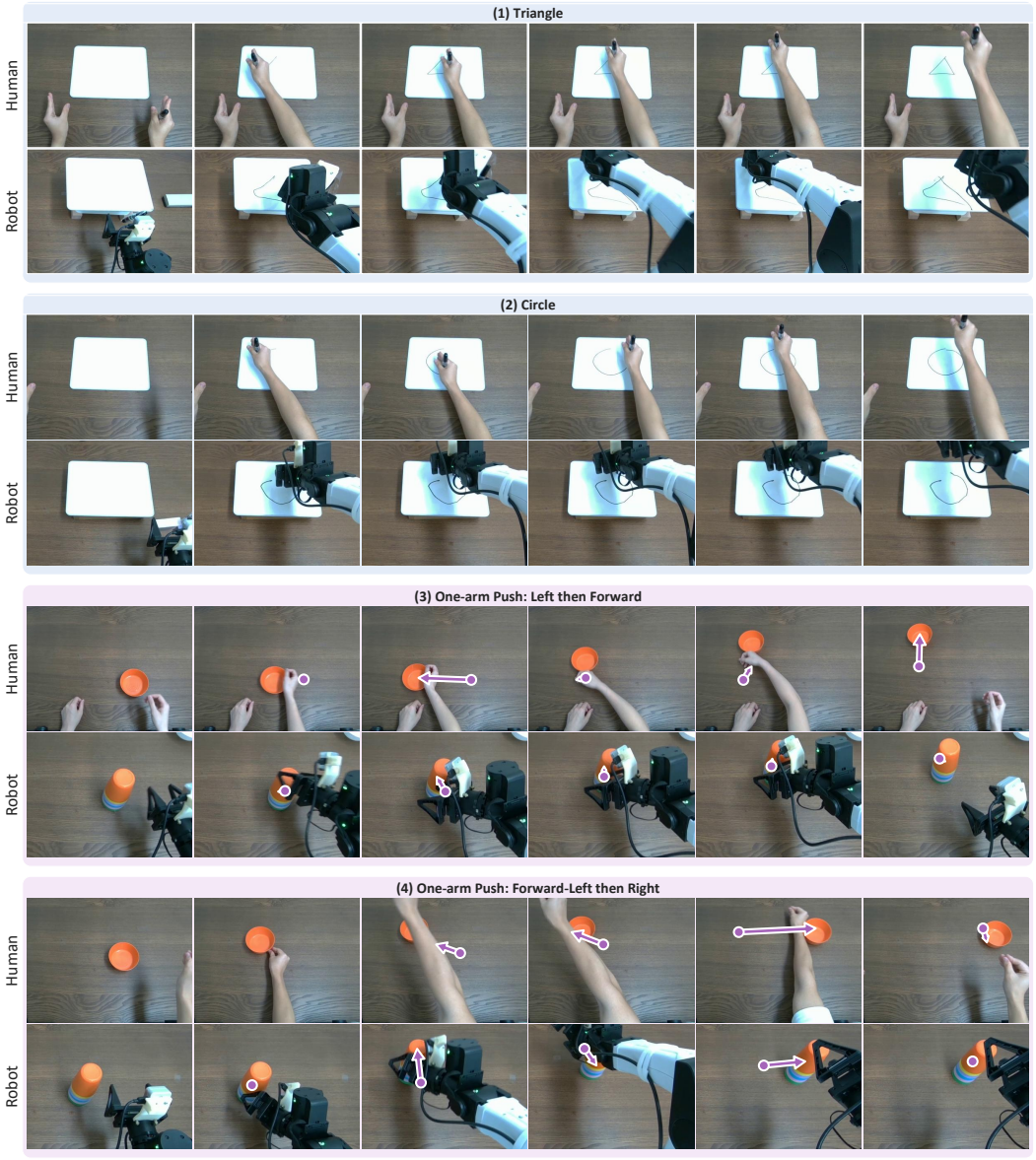


Figure 16: **Additional qualitative human-to-robot action transfer results.** DreamX transfers human demonstrations into executable robot actions while preserving task-specific trajectory structure and temporal ordering across writing and single-arm manipulation examples.

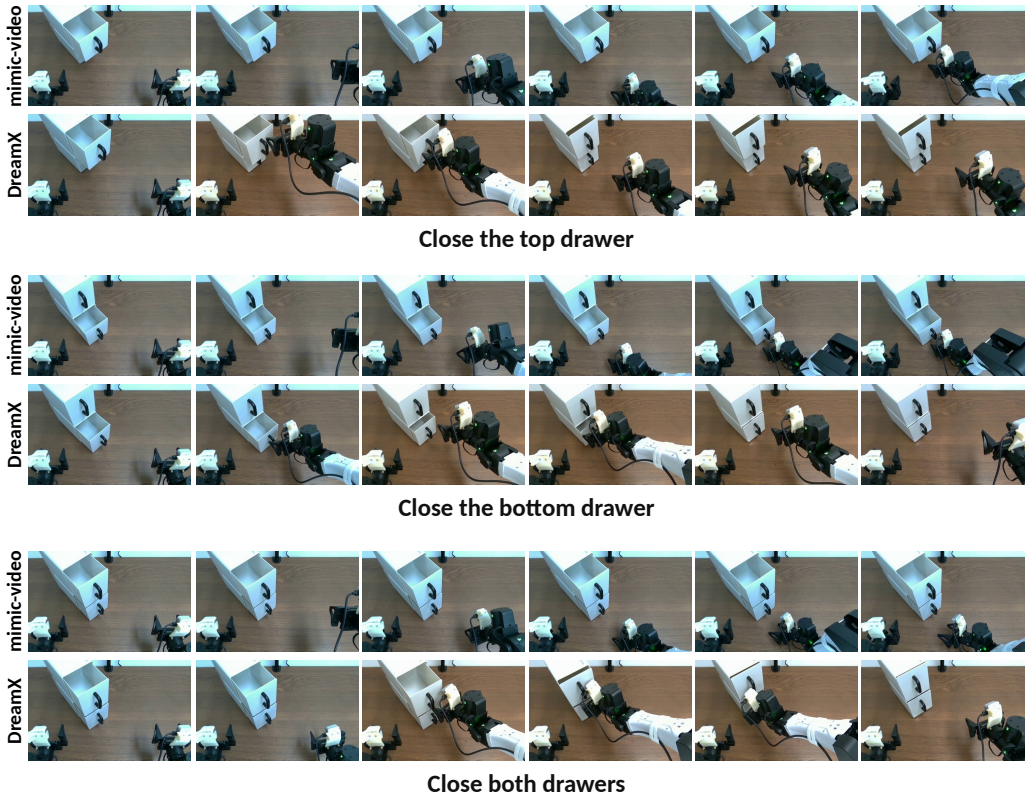


Figure 17: **Real-world drawer-closing rollouts.** Six frames are sampled at equal fractions of each rollout's duration.

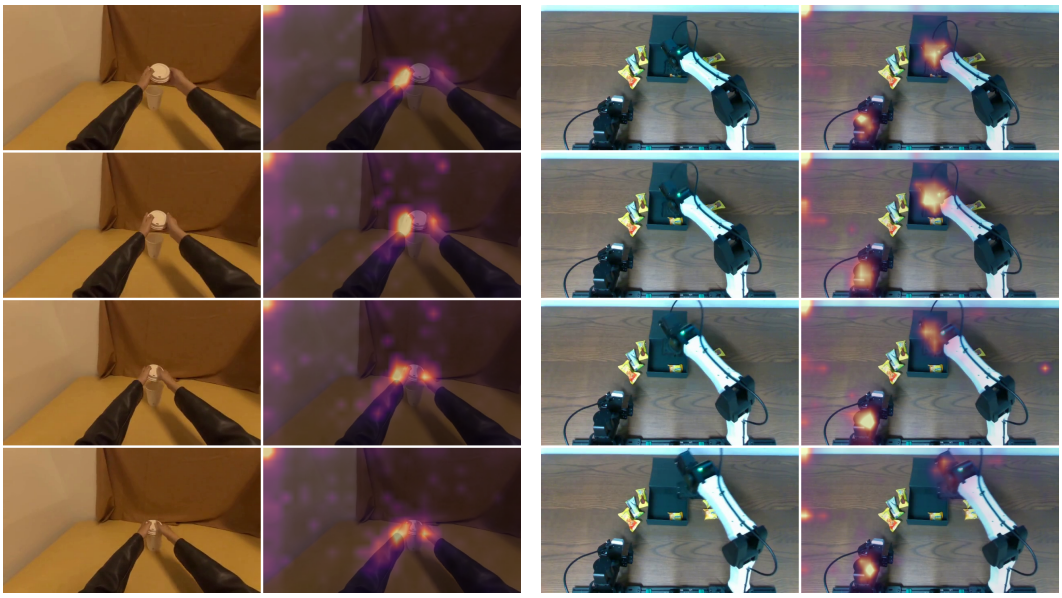


Figure 18: **Query-to-video attention across time.** Human demonstrations (left) and robot demonstrations (right), each shown with the input frame beside its attention overlay. Rows follow temporal order and highlight attention around the acting embodiment and objects.