X-WORLD: REAL-TIME CROSS-EMBODIMENT POLICIES WITH SCALABLE VIDEO WORLD PRIORS

Anonymous authors
Paper under double-blind review

ABSTRACT

Language-instructed manipulation requires robust closed-loop policies, yet action-labeled robot trajectories are costly and limited in coverage. Human and robot videos provide rich spatiotemporal interaction priors that action supervision alone cannot easily capture. Transferring video diffusion priors to policy learning must accommodate variable camera layouts, heterogeneous state and action interfaces, and real-time control. We propose *X-World*, which adapts a pretrained video diffusion backbone to variable-view robot observations, distills its hidden states into a compact token sequence with a Q-Former, and trains a flow-matching action expert that attends to this sequence to predict action chunks. Dataset-specific modules align heterogeneous state and action interfaces. We first adapt the video diffusion backbone on large-scale robot and human videos, freeze it while training the Q-Former and action expert with an action-only objective, and then post-train on the target platform. At inference, *X-World* maintains a 20 Hz model response rate and uses asynchronous real-time chunking to decouple model updates from action execution, avoiding waiting latency. It shows strong performance on in-distribution LIBERO, zero-shot LIBERO-Plus distribution shifts, and real-world manipulation tasks.

1 INTRODUCTION

Language-instructed robotic manipulation Brohan et al. (2022; 2023) requires policies that can translate high-level instructions into robust low-level actions under diverse contacts, partial observability, and long-horizon interactions. While recent vision-language-action (VLA) policies Brohan et al. (2022; 2023); Kim et al. (2024); Octo Model Team et al. (2024); Bjorck et al. (2025); Black et al. (2024); Intelligence et al. (2025) offer an appealing end-to-end recipe, they critically depend on large collections of action-labeled trajectories, which are expensive to acquire and often limited in task, embodiment, and viewpoint coverage. In contrast, internet-scale videos, spanning both human activities and robot interactions, are comparatively abundant and naturally capture rich spatiotemporal regularities such as dynamics and contact events. This gap motivates a central question: how can we leverage scalable video world priors to improve closed-loop manipulation without requiring equally scalable action supervision?

Prior attempts Ye et al. (2024); Du et al. (2023b); Hu et al. (2024) to transfer pretrained video diffusion models to robot policy learning suggest that video world priors can benefit action generation, but practical deployment remains challenging because many methods assume single-view observations or fixed camera layouts, while real-world systems often involve a variable number of cameras. Recent multi-view efforts Liao et al. (2025) often rely on non-seamless backbone modifications, which can shift the model away from its pretrained distribution and weaken transfer. Scaling to heterogeneous datasets O'Neill et al. (2024); Bu et al. (2025a); Wu et al. (2025); Jiang et al. (2025); Wu et al. (2024) is also difficult because different datasets define state and action in different coordinate frames, units, and control conventions. As a result, directly mixing such data under a single action space often yields incompatible supervision signals and destabilizes learning. Finally, expressive generative policies can be expensive at test time, either due to iterative sampling or large backbones, which conflicts with real-time control.

To address these limitations, we propose *X-World*, which (i) adapts a pretrained video diffusion backbone (LTX-Video HaCohen et al. (2024)) to variable-view robot observations via view embed-

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.

dings and a zero-initialized cross-view attention module, (ii) distills diffusion hidden states into a compact token sequence via a Q-Former as a world-prior interface for action generation, (iii) scales action learning across heterogeneous datasets with dataset-specific state and action projection heads that align mismatched state and action spaces while sharing a common action backbone, and (iv) supports continuous control via asynchronous real-time chunking that decouples inference updates from action execution to avoid waiting latency. We train *X-World* in three steps to separate learning a transferable video world prior from learning control. First, we adapt the pretrained video diffusion backbone on large-scale robot O'Neill et al. (2024); Bu et al. (2025a); Wu et al. (2025); Jiang et al. (2025); Wu et al. (2024) and human videos Grauman et al. (2022); Zhao et al. (2025b) to acquire a strong spatiotemporal prior. Then, we freeze the video diffusion backbone and optimize the Q-Former and action expert with the action-only objective $\mathcal{L}_{act}$ for knowledge isolation, so action gradients do not perturb the learned prior. Finally, we post-train on target-platform data to adapt the policy for benchmark or real-world deployment.

We evaluate *X-World* on in-distribution LIBERO Liu et al. (2023), zero-shot LIBERO-Plus Fei et al. (2025) distribution shifts, and real-world manipulation tasks, using task success rate as the primary metric. On LIBERO, *X-World* achieves a 98.5% average success rate, surpassing X-VLA Zheng et al. (2025b) (98.1%), $\pi_{0.5}$ Intelligence et al. (2025) (96.9%), and Genie-Envisioner Liao et al. (2025) (96.5%). Without further adaptation, the same LIBERO-post-trained checkpoint reaches a 76.5% total score on LIBERO-Plus, exceeding OpenVLA-OFT Kim et al. (2025) (69.6%), the strongest listed zero-shot baseline. On real-world evaluation, *X-World* reaches an 81.7% mean success rate across six dual-arm manipulation tasks, consistently outperforming all baselines in our comparison, including Genie-Envisioner Liao et al. (2025), $\pi_{0.5}$ Intelligence et al. (2025), and X-VLA Zheng et al. (2025b). Moreover, with asynchronous real-time chunking that decouples inference updates from action execution, *X-World* enables closed-loop control without waiting latency, sustaining a model update rate of more than 20 Hz on a single H100 GPU, with 46 ms policy inference time.

Our contributions are threefold:

- **Seamless variable-view video diffusion adaptation:** upgrade a pretrained video diffusion backbone to variable-view robot videos via view embeddings and a zero-initialized cross-view attention module.
- **Compact world priors for scalable action generation:** distill diffusion hidden states into a token sequence via a Q-Former, and train a flow-matching action expert that uses the token sequence with dataset-specific modules to align heterogeneous state and action interfaces.
- **Three-stage training with knowledge isolation and low-latency deployment:** train in three stages by adapting the video diffusion backbone on large-scale videos and learning action generation with the backbone kept frozen for knowledge isolation, and deploy with asynchronous real-time chunking for low-latency control.

2 Related work

Video-action models.

Video-action models bridge visual dynamics and control by leveraging world priors inherited from video foundation models for action generation. Recent advancements in this domain Hu et al. (2024); Liao et al. (2025); Li et al. (2026) have demonstrated capabilities comparable to vision-language-action (VLA) policies across a broad spectrum of tasks. Furthermore, in evaluations targeting generalization and zero-shot task completion, these models have been reported to match or even surpass VLA policies in specific settings Ye et al. (2026). Early frameworks Du et al. (2023b;a); Wen et al. (2024) typically adopt a two-stage design, where they first generate future observations as explicit video rollouts and subsequently employ inverse dynamics to map predicted state transitions back to actions. While this pipeline renders the world rollout explicit, prediction errors can propagate to the downstream control module, and the decoupled architecture often increases engineering complexity for closed-loop deployment. To mitigate these issues, recent works extract internal generative features, such as denoising hidden states or latent representations, to serve as world priors for action generation. Certain methods Hu et al. (2024); Yang et al. (2025) compress these features

into compact tokens for cross-attention with an action expert, whereas others Bi et al. (2025); Liao et al. (2025); Ye et al. (2026); Li et al. (2026); Pai et al. (2025) jointly model video and action sequences using mixture-of-transformers architectures. In parallel, Cosmos Policy Kim et al. (2026) represents actions as image-like tokens and reuses video generators to synthesize action sequences directly, offering an alternative that avoids a separate action expert. In contrast, *X-World* adapts a pretrained video diffusion backbone to variable-view robot videos and bypasses explicit video rollouts by compressing its per-layer denoising hidden states into a token sequence via a layer-aligned Q-Former, while following a three-phase training pipeline to better preserve the video model priors.

CROSS-EMBODIMENT TRAINING.

The emergence of robotic foundation models has been significantly propelled by scaling training across expansive datasets from diverse platforms O'Neill et al. (2024). However, naive aggregation of trajectories from disparate embodiments often yields diminishing returns, primarily due to inherent discrepancies in action interfaces and observation spaces. A burgeoning line of work Doshi et al. (2024); Wang et al. (2024); Zheng et al. (2025a;c) addresses these heterogeneities by introducing alignment layers or adapters that project dataset-specific signals into a canonical representation. Beyond action-state mismatches, observation heterogeneity arising from varied camera extrinsics and robot morphologies poses additional challenges. For instance, VLA-LPAF Bian et al. (2025) alleviates viewpoint-induced shifts via representation-level feature alignment, while other strategies operate at the data level through novel-view synthesis Chen et al. (2024b) or embodiment inpainting Chen et al. (2024a;b); Ji et al. (2025). Distinctively, *X-World* capitalizes on large-scale video pretraining to internalize world dynamics across diverse observations, while concurrently bridging heterogeneous action interfaces via modular adapters to facilitate unified policy learning.

3 METHOD

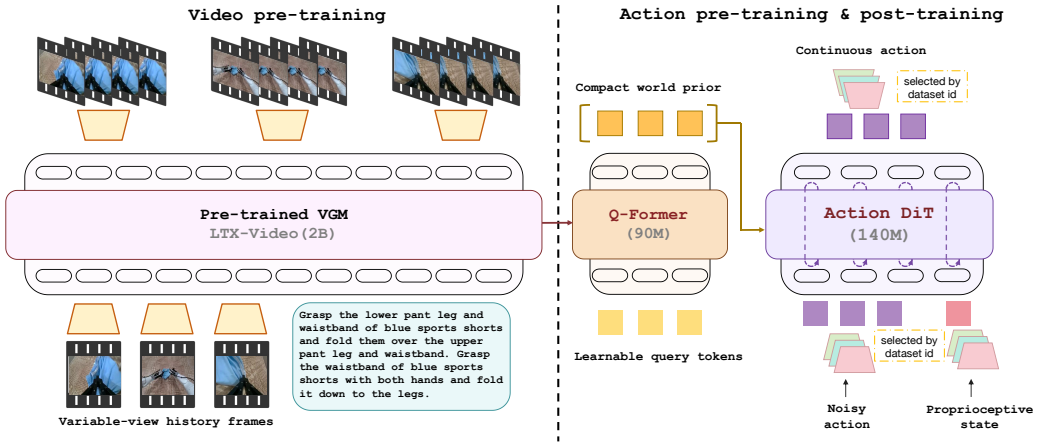
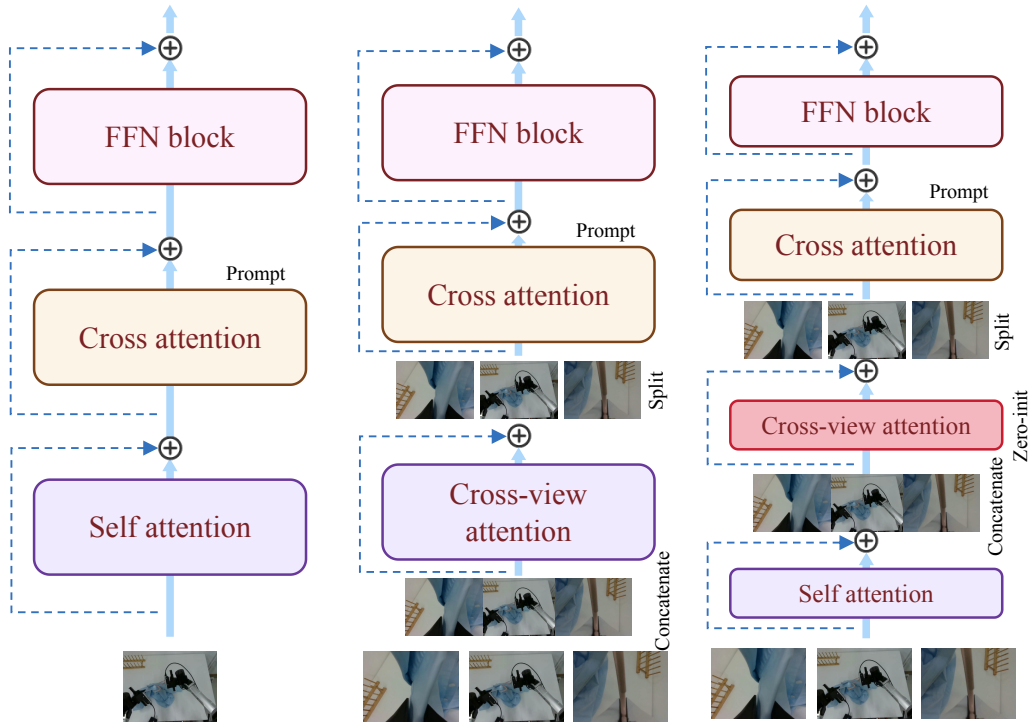


Figure 1: Left: Video pretraining. A pretrained video diffusion backbone learns variable-view world priors from large-scale robot and human videos. **Right: Action pre-training and post-training.** A Q-Former compresses backbone hidden states into a compact world prior for the flow-matching action expert.

3.1 OVERVIEW

We study language-instructed closed-loop manipulation where, at time step t, the policy takes variable-view RGB observations $\mathcal{I}_t = \{I_t^{(v)}\}_{v=1}^{V_t}$, a language instruction l, and proprioceptive state s_t, and outputs a short action chunk in the embodiment's native action space. In §3.2, we introduce *X-World*, which seamlessly adapts a pretrained video diffusion model to robot observations and leverages its world prior to scale action learning on heterogeneous VLA datasets. In §3.3, we describe a three-stage training pipeline that learns the video world prior from videos and transfers it to action generation. Finally, in §3.4, we present asynchronous real-time chunking for low-latency closed-loop control. An overview of the full pipeline is shown in Fig. 1.

3.2 MODEL ARCHITECTURE



(a) LTX-Video (b) Genie-Envisioner (c) X-World (Ours)

Figure 2: DiT block comparison. (a) LTX-Video HaCohen et al. (2024) is single-view. (b) Genie-Envisioner Liao et al. (2025) replaces self-attention with three-view cross-view attention. (c) *X-World* retains self-attention with a zero-init residual cross-view module for seamless transfer.

X-World consists of three modules: (i) a variable-view video diffusion backbone that encodes spatiotemporal world representations, (ii) a Q-Former that distills them into a compact token sequence serving as a world prior, and (iii) a shared action expert that attends to this world prior to generate action chunks, with dataset-specific modules aligning state and action interfaces across datasets.

VIDEO DIFFUSION BACKBONE WITH SEAMLESS VARIABLE-VIEW ADAPTATION.

We start from a pretrained LTX-Video diffusion backbone HaCohen et al. (2024) and adapt it to variable-view robot observations. Given an instruction l and a *single* variable-view history frame at time t, the backbone is trained to predict a 256×256 future clip over a video-adaptation horizon of $K{=}24$, distinct from the executable $H{=}16$ action chunk used for closed-loop control. We choose LTX-Video because its video VAE provides $16\times$ spatial downsampling and $8\times$ temporal downsampling, reducing video token count and enabling efficient training and inference.

Handling multiple camera views is necessary in practice as different scenes can provide different numbers of cameras and viewpoints. Genie-Envisioner Liao et al. (2025) extends LTX-Video to three views by replacing pretrained self-attention with cross-view attention, concatenating tokens across views before attention and splitting them afterward. However, this modification changes the backbone function at initialization and can shift it away from the pretrained distribution. Moreover, Genie-Envisioner is trained on the three-view AgiBotWorld dataset Bu et al. (2025a), constraining downstream policies to a three-view observation format.

To preserve the pretrained behavior at initialization while enabling cross-view fusion under variable-view inputs, we keep the pretrained self-attention unchanged and append an additional cross-view attention module as a zero-initialized residual branch (Fig. 2). We also add a view embedding to encode view identity without assuming a fixed semantic ordering, and we randomize view order during training to improve robustness to heterogeneous camera setups. During training, we represent each training sample with three view slots. When only one or two views are available in a dataset,

we replicate an existing view to fill the missing slots and apply independent diffusion noise to each replica, which avoids padding and prevents wasting computation on masked placeholders. To ensure the replicated views do not affect the original views and vice versa, we apply cross-view attention masks that block any attention between replicated and original view slots. For different video frame rates, we provide per-sample fps to scale temporal RoPE interpolation Su et al. (2024), aligning temporal positions with sampling rates.

Q-FORMER FOR A COMPACT VIDEO WORLD PRIOR.

The video diffusion backbone produces rich spatiotemporal features throughout the denoising process. We denote by $\{h_\ell\}_{\ell=1}^{L}$ the hidden states exported from its L denoising transformer blocks. Naively exposing all layers and tokens to the action model Liao et al. (2025); Intelligence et al. (2025) yields a large cross-attention memory and leaves the action expert to fuse multi-layer information, increasing optimization difficulty and underusing deeper layers because late denoising features are accessible only through the deepest cross-attention layers. We instead distill $\{h_\ell\}$ into a fixed-length token sequence, giving the action expert an efficient compact world prior. Specifically, our Q-Former is layer-aligned with the video backbone. It refines learnable query tokens through a stack of L cross-attention blocks. At layer ℓ, the current tokens $z_{\ell-1}$ cross-attend to the corresponding backbone hidden states h_ℓ and update to z_ℓ. We take the final token sequence $z \triangleq z_L$ as the compact world prior. This fixed-length representation is independent of the number of input views and video tokens, so the action expert attends to a fixed interface rather than a variable-size video feature set.

ACTION EXPERT WITH DATASET-SPECIFIC ALIGNMENT.

VLA datasets have heterogeneous state and action interfaces, including action parameterizations, units, and coordinate frames. Rather than forcing a single shared action space, inspired by X-VLA Zheng et al. (2025b), we decouple dataset-specific interfaces from a shared action-generation backbone via lightweight dataset-specific modules. Each dataset has a state projector, an action input projection for noised action chunks, and an output head that decodes shared representations into executable actions in its native parameterization. These dataset-specific modules only affect the input and output layers, allowing the shared backbone to scale across heterogeneous state and action interfaces without imposing a single unified action space. The shared action expert is a DiT-style transformer Peebles & Xie (2023) that operates on the concatenated state and action tokens. It interleaves self-attention over the action-trajectory tokens with cross-attention to the compact world prior z exported by the Q-Former.

3.3 TRAINING

We train *X-World* in three stages: (1) world prior adaptation, (2) action expert pretraining, and (3) joint post-training on the target-platform dataset.

STAGE 1: WORLD PRIOR ADAPTATION.

We adapt the video diffusion backbone on robot and human videos by optimizing a future-prediction objective in the VAE latent space. Following standard flow matching, we sample a noise level σ from the scheduler and draw $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$. For a clean future latent chunk $\mathbf{y}$, we construct a noised latent $\mathbf{y}_\sigma = (1-\sigma)\mathbf{y} + \sigma\boldsymbol{\epsilon}$ and regress the target vector field $(\boldsymbol{\epsilon} - \mathbf{y})$:

$$\mathcal{L}_{\text{vid}} = \mathbb{E}\Big[\big\|v_\phi(\mathbf{y}_\sigma, \sigma, \mathcal{I}_t, l) - (\boldsymbol{\epsilon} - \mathbf{y})\big\|_2^2\Big]. \tag{1}$$

To support classifier-free guidance, we apply caption dropout by randomly replacing the instruction with an empty prompt embedding. We use the variable-view training protocol described in §3.2 with view-order randomization, view replication for missing view slots, cross-view attention masking, and per-sample fps input for temporal RoPE scaling. We train on large-scale open-source robot and human video datasets (see Supplement). We train this stage for 7 days on 64 H100 GPUs.

STAGE 2: ACTION EXPERT PRETRAINING.

In Stage 2, we freeze the video diffusion backbone and train the Q-Former, the shared action expert, and dataset-specific modules on heterogeneous VLA datasets with action supervision, so this training does not perturb the world prior from Stage 1. To mitigate dominance from large datasets, we sample training batches using a temperature-scaled mixing scheme to increase the sampling probability of smaller datasets. Following standard flow matching, we sample an action noise level σ_a and draw $\boldsymbol{\epsilon}_a \sim \mathcal{N}(0, \mathbf{I})$. For a ground-truth action chunk $\mathbf{a} \in \mathbb{R}^{H \times d_a}$ with H=16, we construct a noised chunk $\mathbf{a}' = \mathbf{a} + \sigma_a(\boldsymbol{\epsilon}_a - \mathbf{a})$ and regress the target vector field $(\boldsymbol{\epsilon}_a - \mathbf{a})$:

$$\mathcal{L}_{\text{act}} = \mathbb{E}\Big[\big\|v_\theta(\mathbf{a}', \sigma_a, z, s_t) - (\boldsymbol{\epsilon}_a - \mathbf{a})\big\|_2^2\Big], \tag{2}$$

where d_a is the action dimension and s_t is the proprioceptive state. We train on large-scale open-source robotics VLA datasets (see Supplement) for 3 days on 64 H100 GPUs.

STAGE 3: JOINT POST-TRAINING.

Finally, we post-train on self-collected real-world teleoperation trajectories to adapt *X-World* to the target platform. We update all model parameters and optimize a joint objective that combines the video and action losses to adapt both the video world prior and action generation to the target domain:

$$\mathcal{L}_{\text{post}} = \mathcal{L}_{\text{vid}} + \mathcal{L}_{\text{act}}. \tag{3}$$

3.4 REAL-TIME INFERENCE

Closed-loop manipulation requires low-latency updates while maintaining a continuous action stream. At control time, we VAE-encode the current views, keep their condition latents fixed, and initialize unobserved future latent slots with noise. The policy then performs one LTX forward pass at the highest-noise endpoint (σ=1.0, t=1000), with condition-frame tokens masked to timestep zero, to extract hidden states. The Q-Former and a four-step flow action expert predict actions without iterative future-video denoising or decoding. Inspired by RTC Black et al. (2025a;b), we therefore use asynchronous real-time chunking to decouple inference updates from action execution: the robot always executes committed actions, while the policy refreshes future actions in the background.

We maintain a FIFO action queue $\mathcal{Q}$ and execute and remove its front action at each control step. Let $R \triangleq |\mathcal{Q}|$ be the remaining queue length, P the prefix threshold, and E the number of newly predicted suffix actions appended after each refresh. When $R \leq P$, the policy uses the current observations and reuses the remaining R actions as a committed prefix to generate a refreshed action chunk. It discards the overlapping R predicted actions and appends the next E suffix actions to $\mathcal{Q}$. Choosing P so that the execution time of P control steps exceeds the end-to-end update latency keeps the queue non-empty while the next suffix is generated in the background. This avoids waiting gaps and maintains continuity at the chunk boundary.

During Stage 2 and Stage 3, we mirror this deployment interface by uniformly sampling an overlap length $\delta \sim \text{Uniform}(\{0, \ldots, \delta_{\max}\})$ with $\delta_{\max}$=8. We keep the first δ prefix steps noise-free by setting their action noise levels to zero. The resulting per-step action noise level is

$$\sigma_{a,i} = \begin{cases} 0, & i \leq \delta, \\ \sigma_a, & i > \delta, \end{cases} \tag{4}$$

For a ground-truth action chunk $\mathbf{a} \in \mathbb{R}^{H \times d_a}$ and Gaussian noise $\boldsymbol{\epsilon}_a \sim \mathcal{N}(0, \mathbf{I})$, the partially noised action trajectory is $\mathbf{a}'_i = \mathbf{a}_i + \sigma_{a,i}(\boldsymbol{\epsilon}_{a,i} - \mathbf{a}_i)$.

4 EXPERIMENTS

4.1 VIDEO GENERATION

We qualitatively evaluate the adapted video diffusion backbone by visualizing future predictions from a variable-view history frame and a language instruction. Representative predictions after

Table 1: LIBERO success rates (%). **Best** and **second-best** results are highlighted.

Method	Spatial↑	Object↑	Goal↑	Long↑	Average
Diffusion Policy	78.5	87.5	73.5	64.8	76.1
OpenVLA	84.7	88.4	79.2	53.7	76.5
SpatialVLA	88.2	89.9	78.6	55.5	78.1
CoT-VLA	87.5	91.6	87.6	69.0	83.9
π_0-Fast	96.4	96.8	88.6	60.2	85.5
GR00T-N1	94.4	97.6	93.0	90.6	93.9
π_0	98.0	96.8	94.4	88.4	94.4
F1	98.2	97.8	95.4	91.3	95.7
InternVLA-M1	98.0	**99.0**	93.8	92.6	95.9
Discrete Diffusion VLA	97.2	98.6	97.4	92.0	96.3
Genie-Envisioner	98.2	97.6	95.8	94.4	96.5
$\pi_{0.5}$	**98.8**	98.2	**98.0**	92.4	96.9
GR00T-N1.6	97.7	98.5	97.5	94.4	97.0
OpenVLA-OFT	97.6	98.4	97.9	94.5	97.1
X-VLA	98.2	98.6	97.8	**97.6**	**98.1**
X-World	**99.6**	**99.4**	**99.0**	**95.8**	**98.5**

Table 2: LIBERO-Plus zero-shot success rates (%) on the most diagnostic shifts. The complete seven-shift breakdown is in the Supplement.

Method	Robot	Lang.	Light	Layout	Total
OpenVLA-OFT	31.9	79.5	88.7	74.2	69.6
π_0-Fast	21.6	61.0	73.2	68.8	61.6
RIPT-VLA	31.2	77.6	88.4	74.2	68.4
OpenVLA-OFT$_{\text{m}}$	21.7	81.0	92.7	68.7	67.9
X-World	**72.5**	**94.3**	**95.4**	**84.7**	**76.5**

Stage 1 adaptation, together with examples after joint post-training on ∼30 minutes of real-world teleoperation data, are provided in the Supplement. The predictions cover one, two, and three input views and show frames 8, 16, and 24 of the predicted $K{=}24$ future clip, demonstrating rapid adaptation to a new scene.

4.2 Simulation Results

We report simulation results on LIBERO Liu et al. (2023), a benchmark for language-instructed tabletop manipulation organized into four suites: Spatial, Object, Goal, and Long. We post-train a single unified *X-World* policy using LIBERO data from all four suites, and evaluate it on each suite by reporting success rate (%), averaged over 50 episodes for each of the 10 tasks, which amounts to 500 rollouts per suite. For observations, we use two synchronized RGB views, a third-person view and an eye-in-hand wrist view, along with the language instruction. Table 1 compares *X-World* with recent VLA and video-action baselines Chi et al. (2025); Kim et al. (2024); Qu et al. (2025); Zhao et al. (2025a); Pertsch et al. (2025); Bjorck et al. (2025); Black et al. (2024); Lv et al. (2025); Contributors (2025); Liang et al. (2025); Liao et al. (2025); Intelligence et al. (2025); Kim et al. (2025); Zheng et al. (2025b). *X-World* achieves the highest average success rate of 98.5%, reaching 99.6% on Spatial, 99.4% on Object, and 99.0% on Goal. On the Long suite, which stresses long-horizon multi-step execution, *X-World* attains 95.8% success. Overall, *X-World* performs strongly across all four suites, suggesting that transferring the video world prior to action generation improves robustness across diverse task compositions and horizons.

LIBERO-Plus generalization. We further evaluate zero-shot generalization on LIBERO-Plus Fei et al. (2025), which assesses robustness to shifts in camera, robot, language, lighting, background, observation noise, and object layout. We evaluate the checkpoint post-trained on LIBERO without further adaptation to LIBERO-Plus. We compare against existing zero-shot baselines Kim et al. (2024); Black et al. (2024); Kim et al. (2025); Hung et al. (2025); Cen et al. (2025); Bu et al. (2025b); Pertsch et al. (2025); Tan et al. (2025). As shown in Table 2, *X-World* reaches a total score

of 76.5%, exceeding OpenVLA-OFT (69.6%), the strongest zero-shot baseline. It is particularly robust to robot, language, lighting, and layout shifts.

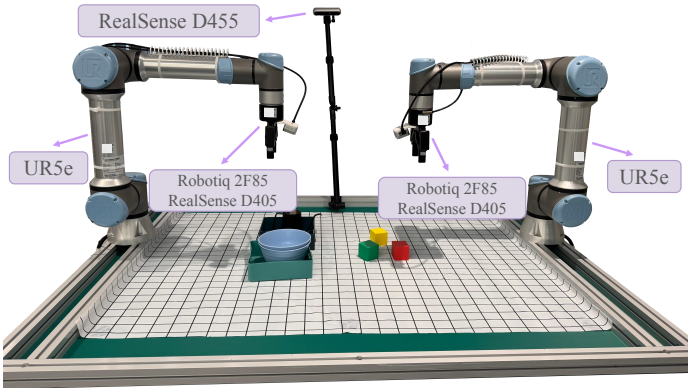


Figure 3: Dual-arm UR5e setup with multi-view cameras.

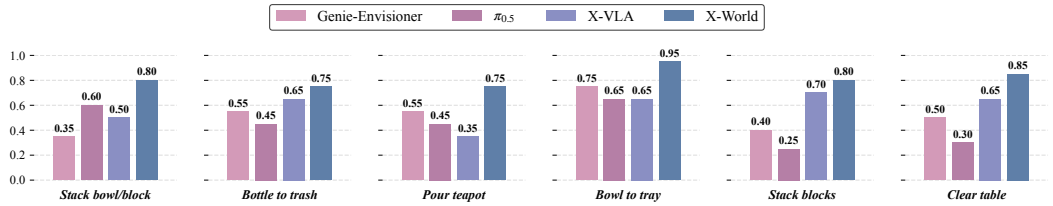


Figure 4: Real-world success rates on a dual-arm UR5e platform over 20 trials per task.

4.3 Real-world results

We evaluate *X-World* on a dual-arm UR5e platform, with the robot configuration and multi-view camera setup shown in Fig. 3. Our platform uses two UR5e arms, each with a Robotiq 2F85 gripper and a wrist-view Intel RealSense D405 camera. We also mount an overhead third-person Intel RealSense D455 camera to provide a global view of the workspace. For each task, we collect 30 minutes of VR teleoperation data and post-train the policy on the collected trajectories. We record data and run closed-loop control at 30 Hz. For execution, we represent actions as relative joint position targets and include absolute joint positions as proprioceptive state inputs. We evaluate 20 trials per task, each using a fixed initial placement from a set of 20 predefined positions. We evaluate six bimanual tasks: *Stack bowl and block*, *Put bottle into trash can*, *Pour water from teapot*, *Move bowl into a tray*, *Stack two blocks*, and *Clear the table*. They require sequential and simultaneous coordination. Task descriptions are provided in the Supplement. We additionally evaluate a long-horizon *Fold Towel* task, which requires first flattening a cluttered towel and then folding it. *X-World* succeeds in 12 of 20 trials.

Figure 4 reports *X-World* and baseline success rates under a unified protocol Liao et al. (2025); Intelligence et al. (2025); Zheng et al. (2025b). *X-World* achieves the best success rate on all six tasks. It attains 75%–95% success with an average of 81.7%. X-VLA Zheng et al. (2025b) averages 58.3%, Genie-Envisioner Liao et al. (2025) 51.7%, and $\pi_{0.5}$ Intelligence et al. (2025) 45.0%. Representative rollouts and qualitative recovery behavior are provided in the Supplement.

4.4 Ablation studies

Training-stage. We study each training stage by removing Stage 1 (video prior adaptation) or Stage 2 (action expert pre-training), while keeping the LIBERO Liu et al. (2023) post-training and evaluation fixed. (i) w/o Stage 1: we skip video prior adaptation, initialize from pretrained LTX-Video HaCohen et al. (2024), and perform Stage 2 action expert pre-training. (ii) w/o Stage 2: we remove action expert pre-training, and train the policy on LIBERO without the large-scale heterogeneous VLA pretraining stage. (iii) w/o Stage 1 and Stage 2: we initialize from the pretrained

Table 3: LIBERO ablations.

Setting	Spatial↑	Object↑	Goal↑	Long↑	Average
w/o Stage 1	95.4	97.6	92.4	72.4	89.5
w/o Stage 2	99.2	98.8	97.2	94.2	97.4
w/o Stage 1 and Stage 2	97.4	96.2	93.4	81.2	92.1
Shared projections	98.6	98.4	98.2	94.4	97.4
w/o Q-Former	**99.8**	98.6	**98.6**	94.6	97.9
w/o video loss	**99.8**	**100.0**	97.2	**96.2**	**98.3**
Full model	**99.6**	**99.4**	**99.0**	**95.8**	**98.5**

LTX-Video backbone HaCohen et al. (2024) and post-train the policy on LIBERO directly, without Stage 1 and Stage 2. As reported in Table 3, the full model achieves the best average success rate (98.5%). Removing Stage 1 causes the largest degradation, dropping the average to 89.5% and the Long suite to 72.4%, which indicates that video prior adaptation is critical for long-horizon robustness. Removing Stage 2 leads to a smaller but consistent drop (98.5% to 97.4%), and removing both stages further reduces the average to 92.1%.

Q-Former. We further ablate the compact token sequence by removing the Q-Former and letting the action expert attend directly to the last-layer hidden states exported by the video diffusion backbone. This variant reaches 97.9% average, below the full model (98.5%), suggesting that the compact token improves overall balance across suites. The Q-Former exposes a fixed-length interface to the action expert and costs 5 ms within the profiled model response, avoiding direct action-expert cross-attention whose context length grows with the video features.

Variable-view design. Disabling the residual cross-view attention reduces average success to 83.1%, while removing the view embedding reduces it to 93.3% (vs. 98.5% for the full model). On Fold Towel, these variants succeed in 0/20 and 2/20 trials, respectively, which supports the roles of cross-view fusion and view identity beyond the near-saturated LIBERO average.

Heterogeneous alignment. Replacing dataset-specific modules with shared projections during Stage 2 lowers the average success rate from 98.5% to 97.4%, with consistent drops across all four suites, supporting their role in aligning heterogeneous state and action.

w/o video loss in post-training. Optimizing only the action loss reaches 98.3% on average (vs. 98.5% for the full model), suggesting that video loss improves balance across suites and may reinforce the transferred prior.

Runtime and real-time chunking analysis. The cached model response takes 46 ms (22 Hz), while the first call including one-time T5 encoding takes 59 ms. With data processing, network transfer, and control-side overhead, the end-to-end refresh takes 91 ms, distinct from the 30 Hz action-execution rate. With P=6, the committed prefix lasts 200 ms and keeps execution continuous while the next suffix is updated in the background; the full timing breakdown is in the Supplement.

5 Conclusion

We presented *X-World*, a world-to-action framework that transfers scalable video world priors to language-instructed closed-loop manipulation. It adapts a pretrained video diffusion backbone to variable-view robot observations, distills its hidden states into a compact token sequence with a Q-Former, and uses a flow-matching action expert to generate executable action chunks. Training freezes the video diffusion backbone during action-only learning, while dataset-specific modules align heterogeneous state and action interfaces. At inference, asynchronous real-time chunking decouples model updates from action execution for continuous control without waiting latency. Experiments on in-distribution LIBERO, zero-shot LIBERO-Plus distribution shifts, and real-world tasks demonstrate strong performance.

AI USE STATEMENT

In preparing this manuscript and the associated submission files, we used generative AI tools, including OpenAI Codex, to assist with manuscript organization, prose polishing, LaTeX formatting, and software debugging. The authors reviewed and revised all AI-assisted text and suggestions, checked technical statements and citations against the source records, and retain responsibility for the final scientific decisions, retained content, and reported results.

ETHICS STATEMENT

This work uses public human and robot video datasets and self-collected robot teleoperation data for research on language-instructed manipulation. Public data are used according to their licenses. Teleoperation recordings are handled as potentially identifying data, and any redistribution should preserve the applicable consent and privacy requirements. Physical robot experiments require supervised operation, workspace safeguards, and an emergency stop; success in simulation or a qualitative rollout does not establish general hardware safety.

REPRODUCIBILITY STATEMENT

The main paper specifies the model components, training objectives, inference queue rule, and evaluation protocols. The Appendix provides the data summaries, implementation details, complete LIBERO-Plus breakdown, ablations, runtime measurements, task definitions, and qualitative evidence needed to reproduce the reported comparisons under the stated hardware and dataset conditions.

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. *arXiv preprint arXiv:2512.13030*, 2025.

Jinyue Bian, Zhaoxing Zhang, Zhengyu Liang, Shiwei Zheng, Shengtao Zhang, Rong Shen, Chen Yang, and Anzhou Hou. Vla-lpaf: Lightweight perspective-adaptive fusion for vision-language-action to enable more unconstrained robotic manipulation. *arXiv preprint arXiv:2509.18183*, 2025.

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

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

Kevin Black, Manuel Y Galliker, and Sergey Levine. Real-time execution of action chunking flow policies. *arXiv preprint arXiv:2506.07339*, 2025a.

Kevin Black, Allen Z Ren, Michael Equi, and Sergey Levine. Training-time action conditioning for efficient real-time chunking. *arXiv preprint arXiv:2512.05964*, 2025b.

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

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

Qingwen Bu, 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.

Jun Cen, Chaohui Yu, Hangjie Yuan, Yuming Jiang, Siteng Huang, Jiayan Guo, Xin Li, Yibing Song, Hao Luo, Fan Wang, Deli Zhao, and Hao Chen. Worldvla: Towards autoregressive action world model. *arXiv preprint arXiv:2506.21539*, 2025.

Lawrence Yunliang Chen, Kush Hari, Karthik Dharmarajan, Chenfeng Xu, Quan Vuong, and Ken Goldberg. Mirage: Cross-embodiment zero-shot policy transfer with cross-painting. *arXiv preprint arXiv:2402.19249*, 2024a.

Lawrence Yunliang Chen, Chenfeng Xu, Karthik Dharmarajan, Muhammad Zubair Irshad, Richard Cheng, Kurt Keutzer, Masayoshi Tomizuka, Quan Vuong, and Ken Goldberg. Rovi-aug: Robot and viewpoint augmentation for cross-embodiment robot learning. *arXiv preprint arXiv:2409.03403*, 2024b.

Tianxing Chen, Zanxin Chen, Baijun Chen, Zijian Cai, Yibin Liu, Zixuan Li, Qiwei Liang, Xianliang Lin, Yiheng Ge, Zhenyu Gu, et al. Robotwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation. *arXiv preprint arXiv:2506.18088*, 2025.

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.

InternVLA-M1 Contributors. Internvla-m1: A spatially guided vision-language-action framework for generalist robot policy. *arXiv preprint arXiv:2510.13778*, 2025.

Ria Doshi, Homer Walke, Oier Mees, Sudeep Dasari, and Sergey Levine. Scaling cross-embodied learning: One policy for manipulation, navigation, locomotion and aviation. *arXiv preprint arXiv:2408.11812*, 2024.

Yilun Du, Mengjiao Yang, Pete Florence, Fei Xia, Ayzaan Wahid, Brian Ichter, Pierre Sermanet, Tianhe Yu, Pieter Abbeel, Joshua B Tenenbaum, et al. Video language planning. *arXiv preprint arXiv:2310.10625*, 2023a.

Yilun Du, Sherry Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Josh Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. *Advances in neural information processing systems*, 36:9156–9172, 2023b.

Senyu Fei, Siyin Wang, Junhao Shi, Zihao Dai, Jikun Cai, Pengfang Qian, Li Ji, Xinzhe He, Shiduo Zhang, Zhaoye Fei, Jinlan Fu, Jingjing Gong, and Xipeng Qiu. Libero-plus: In-depth robustness analysis of vision-language-action models. *arXiv preprint arXiv:2510.13626*, 2025.

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18995–19012, 2022.

Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, et al. Ltx-video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2024.

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.

Chia-Yu Hung, Qi Sun, Pengfei Hong, Amir Zadeh, Chuan Li, U-Xuan Tan, Navonil Majumder, and Soujanya Poria. Nora: A small open-sourced generalist vision language action model for embodied tasks. *arXiv preprint arXiv:2504.19854*, 2025.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. $\pi_{0.5}$: a vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025.

Guanhua Ji, Harsha Polavaram, Lawrence Yunliang Chen, Sandeep Bajamahal, Zehan Ma, Simeon Adebola, Chenfeng Xu, and Ken Goldberg. Oxe-auge: A large-scale robot augmentation of oxe for scaling cross-embodiment policy learning. *arXiv preprint arXiv:2512.13100*, 2025.

Tao Jiang, Tianyuan Yuan, Yicheng Liu, Chenhao Lu, Jianning Cui, Xiao Liu, Shuiqi Cheng, Jiyang Gao, Huazhe Xu, and Hang Zhao. Galaxea open-world dataset and g0 dual-system vla model. *arXiv preprint arXiv:2509.00576*, 2025.

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

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success. In *Robotics: Science and Systems XXI*, 2025. doi: 10.15607/rss.2025.xxi.017.

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.

Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Wensi Ai, Benjamin Martinez, Hang Yin, Michael Lingelbach, Minjune Hwang, Ayano Hiranaka, Sujay Garlanka, Arman Aydin, Sharon Lee, Jiankai Sun, Mona Anvari, Manasi Sharma, Dhruva Bansal, Samuel Hunter, Kyu-Young Kim, Alan Lou, Caleb R Matthews, Ivan Villa-Renteria, Jerry Huayang Tang, Claire Tang, Fei Xia, Yunzhu Li, Silvio Savarese, Hyowon Gweon, C. Karen Liu, Jiajun Wu, and Li Fei-Fei. Behavior-1k: A human-centered, embodied ai benchmark with 1,000 everyday activities and realistic simulation. *arXiv preprint arXiv:2403.09227*, 2024.

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.

Zhixuan Liang, Yizhuo Li, Tianshuo Yang, Chengyue Wu, Sitong Mao, Tian Nian, Liuao Pei, Shunbo Zhou, Xiaokang Yang, Jiangmiao Pang, et al. Discrete diffusion vla: Bringing discrete diffusion to action decoding in vision-language-action policies. *arXiv preprint arXiv:2508.20072*, 2025.

Yue Liao, Pengfei Zhou, Siyuan Huang, Donglin Yang, Shengcong Chen, Yuxin Jiang, Yue Hu, Jingbin Cai, Si Liu, Jianlan Luo, et al. Genie envisioner: A unified world foundation platform for robotic manipulation. *arXiv preprint arXiv:2508.05635*, 2025.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36:44776–44791, 2023.

Qi Lv, Weijie Kong, Hao Li, Jia Zeng, Zherui Qiu, Delin Qu, Haoming Song, Qizhi Chen, Xiang Deng, and Jiangmiao Pang. F1: A vision-language-action model bridging understanding and generation to actions. *arXiv preprint arXiv:2509.06951*, 2025.

Oier Mees, Lukas Hermann, Erick Rosete-Beas, and Wolfram Burgard. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters (RA-L)*, 2022.

Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, et al. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*, 2024.

Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6892–6903. IEEE, 2024.

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.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023.

Karl Pertsch, Kyle Stachowicz, Brian Ichter, Danny Driess, Suraj Nair, Quan Vuong, Oier Mees, Chelsea Finn, and Sergey Levine. Fast: Efficient action tokenization for vision-language-action models. *arXiv preprint arXiv:2501.09747*, 2025.

Delin Qu, Haoming Song, Qizhi Chen, Yuanqi Yao, Xinyi Ye, Yan Ding, Zhigang Wang, JiaYuan Gu, Bin Zhao, Dong Wang, et al. Spatialvla: Exploring spatial representations for visual-language-action model. *RSS*, 2025.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Shuhan Tan, Kairan Dou, Yue Zhao, and Philipp Krähenbühl. Interactive post-training for vision-language-action models, 2025. URL https://arxiv.org/abs/2505.17016.

Lirui Wang, Xinlei Chen, Jialiang Zhao, and Kaiming He. Scaling proprioceptive-visual learning with heterogeneous pre-trained transformers. *Advances in neural information processing systems*, 37:124420–124450, 2024.

Youpeng Wen, Junfan Lin, Yi Zhu, Jianhua Han, Hang Xu, Shen Zhao, and Xiaodan Liang. Vidman: Exploiting implicit dynamics from video diffusion model for effective robot manipulation. *Advances in Neural Information Processing Systems*, 37:41051–41075, 2024.

Kun Wu, Chengkai Hou, Jiaming Liu, Zhengping Che, Xiaozhu Ju, Zhuqin Yang, Meng Li, Yinuo Zhao, Zhiyuan Xu, Guang Yang, et al. Robomind: Benchmark on multi-embodiment intelligence normative data for robot manipulation. *arXiv preprint arXiv:2412.13877*, 2024.

Shihan Wu, Xuecheng Liu, Shaoxuan Xie, Pengwei Wang, Xinghang Li, Bowen Yang, Zhe Li, Kai Zhu, Hongyu Wu, Yiheng Liu, et al. Robocoin: An open-sourced bimanual robotic data collection for integrated manipulation. *arXiv preprint arXiv:2511.17441*, 2025.

Liudi Yang, Yang Bai, George Eskandar, Fengyi Shen, Mohammad Altillawi, Dong Chen, Ziyuan Liu, and Abhinav Valada. Covar: Co-generation of video and action for robotic manipulation via multi-modal diffusion. *arXiv preprint arXiv:2512.16023*, 2025.

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.

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.

Qingqing Zhao, Yao Lu, Moo Jin Kim, Zipeng Fu, Zhuoyang Zhang, Yecheng Wu, Zhaoshuo Li, Qianli Ma, Song Han, Chelsea Finn, et al. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models. In *CVPR*, 2025a.

Zhenyu Zhao, Hongyi Jing, Xiawei Liu, Jiageng Mao, Abha Jha, Hanwen Yang, Rong Xue, Sergey Zakharor, Vitor Guizilini, and Yue Wang. Humanoid everyday: A comprehensive robotic dataset for open-world humanoid manipulation. *arXiv preprint arXiv:2510.08807*, 2025b.

Jinliang Zheng, Jianxiong Li, Dongxiu Liu, Yinan Zheng, Zhihao Wang, Zhonghong Ou, Yu Liu, Jingjing Liu, Ya-Qin Zhang, and Xianyuan Zhan. Universal actions for enhanced embodied foundation models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 22508–22519, 2025a.

Jinliang Zheng, Jianxiong Li, Zhihao Wang, Dongxiu Liu, Xirui Kang, Yuchun Feng, Yinan Zheng, Jiayin Zou, Yilun Chen, Jia Zeng, et al. X-vla: Soft-prompted transformer as scalable cross-embodiment vision-language-action model. *ICLR*, 2025b.

Jinliang Zheng, Jianxiong Li, Zhihao Wang, Dongxiu Liu, Xirui Kang, Yuchun Feng, Yinan Zheng, Jiayin Zou, Yilun Chen, Jia Zeng, et al. X-vla: Soft-prompted transformer as scalable cross-embodiment vision-language-action model. *arXiv preprint arXiv:2510.10274*, 2025c.

A Data and Training Details

This section summarizes the data sources used across the three training stages of *X-World* and the sampling strategy used for heterogeneous action pretraining.

A.1 Stage 1: Video pre-training

We use a mixture of robot videos and human videos to provide diverse visual interaction data for video pre-training. All sources are converted into a unified variable-view format, with up to three views retained for each sample. Table 4 summarizes the resulting data sources.

Table 4: Datasets used in Stage 1 for video pretraining.

Dataset	Views	Summary
RoboTwin 2.0 Chen et al. (2025)	3	Domain-randomized dual-arm benchmark designed for robust bimanual manipulation and transfer.
CALVIN Mees et al. (2022)	2	Long-horizon language benchmark built around static and gripper observations for sequential manipulation.
AgiBotWorld Bu et al. (2025a)	3	Quality-controlled manipulation data collected in realistic settings with broad skill coverage.
Behavior-1K Li et al. (2024)	3	Household simulation benchmark emphasizing realistic environments and long-horizon daily activities.
RoboCOIN Wu et al. (2025)	3	Bimanual multi-embodiment data with hierarchical annotations for coordinated manipulation skills.
Galaxea Jiang et al. (2025)	3	Open-world robot data from everyday scenes with fine-grained subtask language annotations.
Open-X-Embodiment O'Neill et al. (2024)	1-3	Aggregated real-robot data spanning diverse embodiments, camera layouts, and manipulation behaviors.
RoboMIND Wu et al. (2024)	1-3	Unified multi-embodiment teleoperation data covering diverse tasks, objects, and robot platforms.
Ego4D Grauman et al. (2022)	1	Large-scale egocentric daily-life videos capturing diverse activities and environments from a first-person view.
Humanoid-Everyday Zhao et al. (2025b)	1	Open-world humanoid manipulation data with multimodal sensing for whole-body interaction tasks.

A.2 Stage 2: Action pre-training

We use heterogeneous VLA trajectories collected from multiple benchmarks and robot platforms for action pre-training. These datasets differ in embodiment, observation setup, and heterogeneous state and action interfaces, providing broad supervision across diverse manipulation settings. Table 5 summarizes the main VLA data sources.

A.3 Temperature-scaled multi-dataset sampling

To avoid over-sampling the largest sources while still accounting for dataset scale, we use temperature-scaled sampling when drawing training examples from the Stage 2 mixture. The sampling weight of dataset d is defined as

$$w_d \propto \alpha_d \cdot N_d^{1/T}, \tag{5}$$

where N_d denotes the total number of dataset steps, i.e., the number of sampleable state and action transitions in dataset d, α_d is the base mixture weight, and T is the sampling temperature. We set $\alpha_d = 1.0$ and use $T = 2.0$ for all datasets, so the effective reweighting scales with the square root of the dataset step count before normalization.

A.4 Stage 3: Real-world post-training data

We use self-collected teleoperation data for post-training on the six real-world tasks described in Sec. C.2. We collect 30 minutes of demonstrations per task with synchronized three-view observations. The platform, control interface, evaluation protocol, and task definitions are described in Sec. C.2.

Table 5: Datasets used in Stage 2 for action pretraining.

Dataset	Views	Summary
RoboTwin 2.0 Chen et al. (2025)	3	Domain-randomized dual-arm trajectories for robust bimanual manipulation and transfer.
CALVIN Mees et al. (2022)	2	Long-horizon language-conditioned trajectories with static and gripper observations.
AgiBotWorld Bu et al. (2025a)	3	Large-scale manipulation trajectories collected in realistic settings with diverse skills.
Behavior-1K Li et al. (2024)	3	Household embodied benchmark providing long-horizon trajectories in realistic simulation.
RoboCOIN Wu et al. (2025)	3	Bimanual multi-embodiment trajectories with hierarchical annotations for coordinated skills.
Galaxea Jiang et al. (2025)	3	Open-world manipulation trajectories with fine-grained subtask language supervision.
Open-X-Embodiment O'Neill et al. (2024)	1-3	Aggregated cross-embodiment trajectories covering diverse robots, scenes, and tasks.
RoboMIND Wu et al. (2024)	1-3	Unified multi-embodiment teleoperation trajectories across diverse tasks and platforms.

B Implementation Details

This section provides additional implementation details for variable-view processing, the token sequence interface, heterogeneous state and action alignment, and asynchronous real-time chunking.

B.1 Variable-view video processing

We add a view embedding to encode view identity without assuming a fixed semantic ordering, and randomize view order during training to improve robustness to heterogeneous camera setups. Each training sample is represented with three view slots. When only one or two views are available, we replicate an existing view to fill the missing slots and apply independent diffusion noise to each replica, which avoids padding and prevents wasting computation on masked placeholders. Cross-view attention masks block interactions between replicated and original view slots, ensuring that replicated views do not affect the original views and vice versa.

B.2 FPS-aware temporal RoPE scaling

Our video diffusion backbone builds on LTX-Video HaCohen et al. (2024) and ingests videos collected at different frame rates. To align temporal positional encoding with the physical sampling rate of each sample, we pass the per-sample frame rate fps to the backbone and use it to construct the RoPE interpolation tuple Su et al. (2024) in both training and inference:

$$\mathrm{rope_interpolation_scale} = \left(\frac{r_t}{\mathrm{fps}},\ r_s,\ r_s\right), \tag{6}$$

where r_t and r_s denote the temporal and spatial compression ratios of the LTX-Video VAE. For multi-view inputs, the same sample-level fps is repeated across views. In the transformer, this tuple scales the temporal, height, and width coordinates before rotary embedding is applied, so clips with different frame rates share the same architecture while their temporal coordinates remain aligned with the actual sampling rate.

B.3 Q-Former and token sequence interface

The video diffusion backbone produces spatiotemporal features throughout its denoising transformer blocks. We denote the exported hidden states by $\{h_\ell\}_{\ell=1}^{L}$ and distill them into a fixed-length token sequence using a layer-aligned Q-Former. At layer ℓ, the current query tokens cross-attend to the corresponding backbone hidden states h_ℓ before being passed to the next layer. The final token sequence forms a compact world prior whose size is independent of the number of input views and video tokens, and the action expert attends to this fixed interface when generating action chunks.

B.4 Action representation and dataset-specific modules

VLA datasets have heterogeneous state and action interfaces, including different action parameterizations, units, and coordinate frames. Rather than forcing a single shared action space, we use lightweight dataset-specific modules around a shared action-generation backbone, following the design in X-VLA Zheng et al. (2025b). Each dataset has a state projector, an action input projection for noised action chunks, and an output head that decodes shared representations into executable actions in its native parameterization. For the real-world platform, actions are represented as relative joint position targets, while absolute joint positions are provided as proprioceptive state.

B.5 Asynchronous real-time chunking

We maintain a FIFO action queue $\mathcal{Q}$ and execute and remove its front action at each control step. Let $R \triangleq |\mathcal{Q}|$ be the remaining queue length, P the prefix threshold, and E the number of newly predicted suffix actions appended after each refresh. When $R \leq P$, the policy uses the current observations and reuses the remaining R actions as a committed prefix to generate a refreshed action chunk. It discards the overlapping R predicted actions and appends the next E suffix actions to $\mathcal{Q}$. The robot continues executing committed actions while the policy updates future actions in the background, decoupling inference updates from action execution and avoiding waiting gaps.

In the reference deployment implementation, P corresponds to prefix_steps, E corresponds to execution_steps, and delay is computed as $\min(\text{remain_steps}, \text{prefix_steps})$. Thus, delay gives the number of remaining queued actions reused as the action prefix and dropped from the overlapping part of the returned prediction.

C Additional Experiments

We provide additional video-generation results, detailed real-world evaluation settings and task definitions, and further analyses of asynchronous real-time chunking and variable-view inputs.

C.1 Additional video-generation results

Stage 1 world prior adaptation.

Figure 5 provides additional future predictions after Stage 1 world prior adaptation. Each example comprises a history frame followed by predictions at steps 8, 16, and 24 of the $K{=}24$ future clip.

Real-world post-training.

Figure 6 shows future predictions after post-training on the real-world teleoperation data.

C.2 Real-world evaluation

Platform and evaluation protocol.

We evaluate *X-World* on a dual-arm UR5e platform. Each arm uses a Robotiq 2F85 gripper and a wrist-view Intel RealSense D405 camera, while an overhead Intel RealSense D455 camera provides a third-person view of the workspace. We record data and run closed-loop control at 30 Hz using relative joint position targets, with absolute joint positions provided as proprioceptive state. We evaluate 20 trials per task, each using a fixed initial placement from a set of 20 predefined positions. All evaluated policies use the same camera views, control frequency, initial placements, and task definitions.

Task definitions.

We evaluate six bimanual tasks that require coordinated dual-arm manipulation:

- **Stack bowl and block.** The robot stacks a side bowl onto a center bowl before placing a block into the bowl.
- **Put bottle into trash can.** The robot picks up scattered bottles and places them in a trash can.

- **Pour water from teapot.** The robot pours water into a cup before placing the teapot and cup on a tray.
- **Move bowl into a tray.** The robot moves a bowl to the center with one arm and transfers it into a tray with the other.
- **Stack two blocks.** The robot moves blocks from the table to a tray and stacks them.
- **Clear the table.** The robot places all objects into a drawer.

Together, these tasks require both sequential coordination, with arms acting in turn, and simultaneous coordination, with both arms operating concurrently.

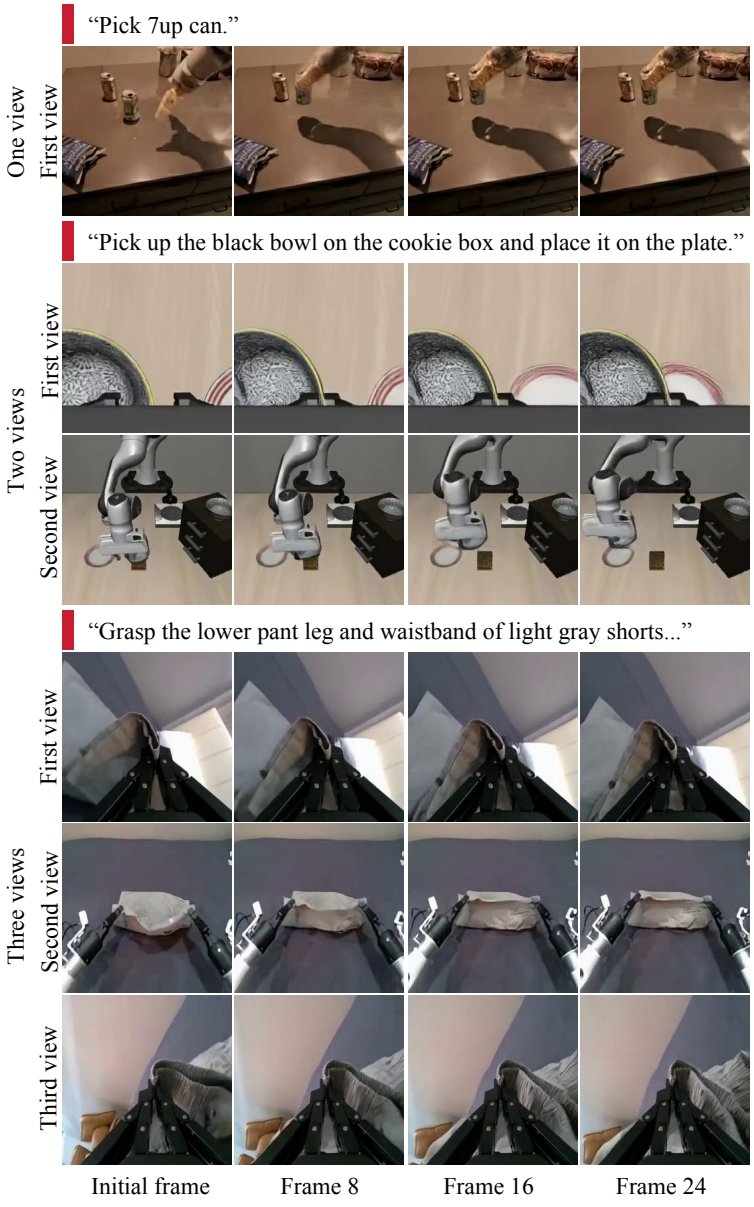


Figure 5: Additional future predictions after Stage 1 world prior adaptation. We show further examples with one, two, and three input views.

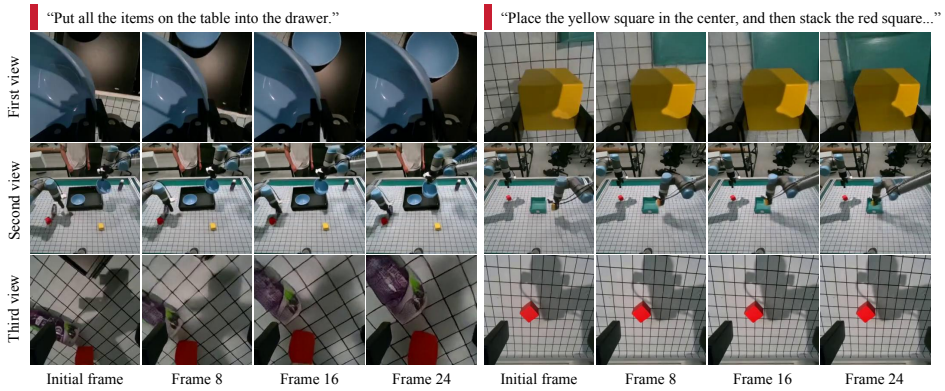


Figure 6: Future predictions after joint post-training on approximately 30 minutes of real-world teleoperation data. Each clip shows the history frame followed by frames 8, 16, and 24 of the predicted $K=24$ future clip.

LONG-HORIZON FOLD TOWEL.

We additionally evaluate a long-horizon *Fold Towel* task, which requires flattening a cluttered towel before completing the fold and therefore tests multi-view coordination over a longer action sequence. The full model succeeds in 12 of 20 trials.

C.3 COMPLETE LIBERO-PLUS BREAKDOWN

The main paper reports the most diagnostic shifts to keep the nine-page presentation readable. Table 6 gives the complete seven-shift comparison used to compute the total score.

Table 6: Complete zero-shot LIBERO-Plus success rates (%) under seven shifts. OpenVLA-OFT+ is excluded because it is trained on LIBERO-Plus.

Method	Cam.	Robot	Lang.	Light	Bg.	Noise	Layout	Total
OpenVLA	0.8	3.5	23.0	8.1	34.8	15.2	28.5	15.6
π_0	13.8	6.0	58.8	85.0	81.4	**79.0**	68.9	53.6
OpenVLA-OFT	**56.4**	31.9	79.5	88.7	**93.3**	75.8	**74.2**	**69.6**
OpenVLA-OFT$_{\text{w}}$	10.4	38.7	70.5	76.8	**93.6**	49.9	69.9	55.8
NORA	2.2	37.0	65.1	45.7	58.6	12.8	62.1	39.0
WorldVLA	0.1	27.9	41.6	43.7	17.1	10.9	38.0	25.0
UniVLA	1.8	**46.2**	69.6	69.0	81.0	21.2	31.9	43.9
π_0-Fast	**65.1**	21.6	61.0	73.2	73.2	74.4	68.8	61.6
RIPT-VLA	55.2	31.2	77.6	88.4	91.6	73.5	**74.2**	68.4
OpenVLA-OFT$_{\text{m}}$	55.6	21.7	**81.0**	**92.7**	91.0	**78.6**	68.7	67.9
X-World	52.6	**72.5**	**94.3**	**95.4**	85.8	59.7	**84.7**	**76.5**

C.4 ABLATIONS AND RUNTIME

Figure 7 summarizes the action-prefix and variable-view evaluations described below.

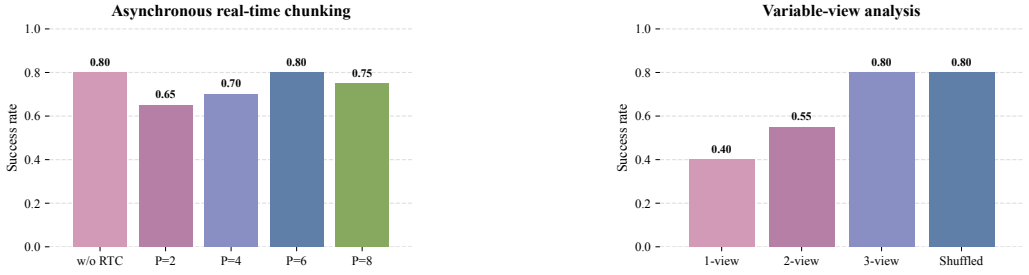


Figure 7: Additional ablations. Left: success rate for asynchronous real-time chunking without RTC and with fixed action prefixes $P \in \{2, 4, 6, 8\}$. **Right:** success rate under one-view, two-view, three-view, and shuffled-view-order inputs using the same model with randomized view order and random view dropping.

ACTION-PREFIX SWEEP.

We study asynchronous real-time chunking with an action-prefix ablation over *w/o real-time chunking* and fixed action prefixes $P \in \{2, 4, 6, 8\}$. As shown in Fig. 7 (left), removing real-time chunking does not substantially change success rate in this setting, but it introduces clearly visible latency during closed-loop execution because policy updates can no longer overlap with action execution. Small action prefixes such as $P{=}2$ remain less reliable because the policy must refresh too frequently, which makes the action queue more vulnerable to transient under-supply. Increasing the action prefix from $P{=}2$ to $P{=}6$ improves success rate, indicating that a moderate overlap better balances continuity and refresh stability. When the action prefix becomes too large, the policy refreshes less frequently, so newly observed changes are reflected more slowly in the executed actions. Based on this trade-off, we use $P{=}6$ in the real-world experiments.

VARIABLE-VIEW ANALYSIS.

We use the same model for all variable-view settings. At evaluation time, the 1-view setting keeps only the third-person camera, the 2-view setting removes one wrist camera, and the shuffled setting uses the same three cameras with their order permuted. As shown in Fig. 7 (right), performance improves substantially from one view to two views and further to three views, indicating that the policy benefits from the additional geometric and occlusion-resolving cues. The one-view and two-view settings retain non-trivial success rates, showing that the model remains functional when partial views are missing. The shuffled setting remains on par with the ordered three-view setting, further confirming that the model does not rely on a fixed camera index at test time.

Using the same LIBERO protocol as the main paper, removing the residual cross-view attention reduces average success from 98.5% to 83.1%, while removing the view embedding reduces it to 93.3%. On Fold Towel, these variants attain 0/20 and 2/20 successes, respectively, compared with 12/20 for the full model.

RUNTIME AND CONTINUITY.

Table 7 reports learned-module and deployment latency. The cached model response takes 46 ms (22 Hz), while the first call including one-time T5 encoding takes 59 ms. With data processing, network transfer, and control-side overhead, the end-to-end refresh takes 91 ms (11 Hz), distinct from the 30 Hz action-execution rate. With $P{=}6$, the committed prefix lasts 200 ms, exceeding refresh latency so actions continue while the next suffix is updated in the background.

Table 7: Runtime breakdown in milliseconds. The first call takes 59 ms. The cached model response includes 14 ms of model-side overhead beyond the listed learned modules.

Model timing		Deployment timing	
T5 encoder (first call)	13	Cached model response	46
VAE	7	Network transfer	32
Video diffusion	12	Data processing	7
Q-Former	5	Control-side overhead	6
Action expert	8	End-to-end refresh	91

D Qualitative Rollouts and Supplementary Videos

The supplementary package includes six real-world rollout videos, one for each task in Sec. C.2. The policy uses synchronized three-view observations, while the videos show the third-person overview for clarity. These examples qualitatively illustrate uninterrupted closed-loop execution; quantitative results for asynchronous real-time chunking are reported in Sec. C.4.

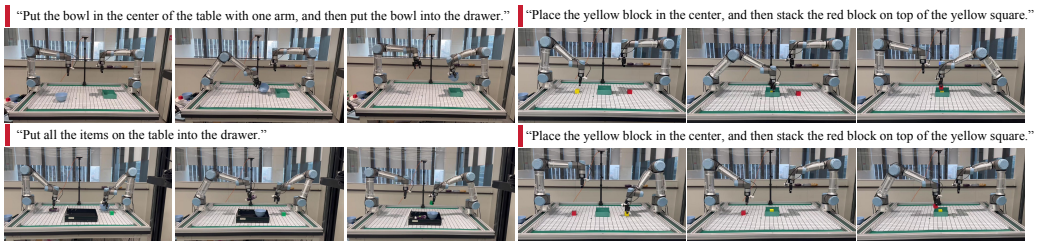


Figure 8: Four representative real-world rollouts on the dual-arm UR5e platform. The displayed sequences are qualitative examples and are not used to compute the success rates.

- 01_stack_bowl_and_block.mp4: *Stack bowl and block*.
- 02_put_bottle_into_trash_can.mp4: *Put bottle into trash can*.
- 03_pour_water_from_teapot.mp4: *Pour water from teapot*.
- 04_move_bowl_into_tray.mp4: *Move bowl into a tray*.
- 05_stack_two_blocks.mp4: *Stack two blocks*.
- 06_clear_the_table.mp4: *Clear the table*.

The videos are qualitative examples and are not used to compute the success rates reported in the main paper or this Supplement.