RETHINKING REPRESENTATION SPACES FOR WORLD-ACTION MODELS: PIXELS, VAE LATENTS, OR SEMANTIC REPRESENTATIONS?

Anonymous authors
Paper under double-blind review

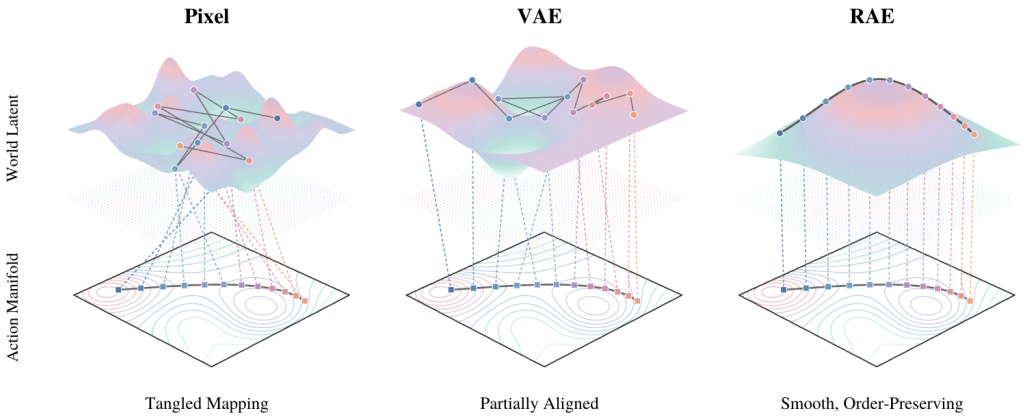

Figure 1: **Representation spaces for world-action models.** We compare pixels, reconstruction-oriented VAE latents, and semantic representations while keeping the future-video prediction task fixed.

ABSTRACT

World-action models (WAMs) built on video generation models learn interaction dynamics that map observations to robot actions. Because video generators can produce implausible or hallucinated futures under perturbation, many recent WAMs add auxiliary supervision on other representations to improve robustness under distribution shift. These additions increase post-training cost and introduce trade-offs among loss weights, while leaving open where generalization comes from. We instead ask a more basic question: *in which representation space should a WAM predict the future?* Within a shared framework and without auxiliary objectives, we compare models that predict the future in pixels (Pixel-WAM), VAE latents (VAE-WAM), or semantic features from frozen pretrained encoders (RAE-WAM). Across six fusion designs, hidden-state fusion transfers video pretraining most effectively. Following this, the V-JEPA RAE-WAM reaches 55.2% success from scratch, compared with 45.7% for the DINOv2 RAE-WAM, 45.4% for VAE-WAM, and 37.2% for Pixel-WAM on LIBERO-Plus. The same representation ordering holds in clean simulated bimanual manipulation, and on a real dual-arm platform the V-JEPA RAE-WAM outperforms $\pi_{0.5}$ and DreamZero. Our results point to a simple principle for building world-action models: rather than reconstructing pixels or relying on auxiliary supervision, predict the future in a pretrained semantic space whose structure already reflects how scenes evolve under action.

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

Robots operating in open environments must anticipate how an action will change the scene, not merely map the current observation and instruction to the next action. World-action models (WAMs) introduce this predictive structure by coupling future-world modeling with action generation, either jointly or through an inverse-dynamics model (Lu et al., 2026). Built on large pretrained video models (Wang et al., 2026a), recent WAMs report improved generalization in novel environments (Ma et al., 2026; Ye et al., 2026; Yan et al., 2026b; Wang et al., 2026d). These results establish the promise of predictive policy learning but not the source of its gains, which may arise from the predictive objective, pretrained backbone, action architecture, or representation in which the future is generated. Separating these factors matters because a visually plausible future is not necessarily useful for control.

We isolate the least studied of these choices: the representation space of the predicted future. Most WAMs inherit VAE latents from video diffusion, whose compression rate and geometry are fixed for pixel reconstruction rather than control. Yet video generators encode more than appearance: their representations transfer to dense tasks such as depth estimation and segmentation (Wang et al., 2026a), and multi-stream WAMs predict several visual modalities in parallel (Yan et al., 2026a). Recent WAMs nevertheless add semantic foresight, latent-dynamics objectives, or parallel representation streams to improve out-of-distribution robustness (Yan et al., 2026b; Chen et al., 2026; Yan et al., 2026a; Liu et al., 2026); the added encoders, heads, and experts confound the contribution of the representation itself. Meanwhile, representation autoencoders and V-RAE show that frozen semantic features can serve directly as generative latent spaces (Zheng et al., 2025; Guo et al., 2026), while JiT-style models remove the tokenizer and generate directly in pixels (Li & He, 2025; Jiang et al., 2026). These alternatives have been evaluated mainly by perceptual generation quality, leaving their utility for closed-loop control unclear.

We therefore ask the question in our title: *should a WAM predict the future in pixels, VAE latents, or semantic representations?* Our hypothesis is that the answer depends on the semantic representation encodes temporal changes aligned with action-relevant dynamics. To test this hypothesis, we hold the future-video prediction task and world-action framework fixed and compare Pixel-WAM, VAE-WAM, and RAE-WAM, with the last instantiated using both DINOv2 and V-JEPA, without auxiliary semantic streams or cross-representation conditioning.

Our contributions are threefold:

- We compare six fusion designs with and without video pretraining, revealing how the fusion design determines whether pretrained features transfer effectively.
- We compare Pixel-WAM, VAE-WAM, and RAE-WAM while keeping other conditions fixed and find out that V-JEPA RAE-WAM, rather than VAE-WAM whose semantic features compress temporal context, is significantly best overall showing that the benefit comes from temporally structured semantic compression.
- We find a clear gain from video pretraining of the RAE-WAM and on a real dual-arm platform, the pretrained V-JEPA RAE-WAM outperforms strong VLA and WAM baselines.

2 RELATED WORK

Representation spaces for visual generation. Image and video diffusion models commonly operate in VAE latents to reduce the spatial and temporal cost of generation. The codec is typically trained for reconstruction and then frozen, making its compression rate, regularization, and latent geometry part of the generative model. Recent analyses argue that reconstruction quality alone does not determine generative performance and instead characterize a latent space by its *diffusability*, including semantic separability and ambiguity of the learned velocity field (Zhong et al., 2026). Representation autoencoders replace reconstruction-oriented VAE features with representations from pretrained vision encoders (Zheng et al., 2025), while V-RAE extends this design to video by adding temporal compression and a video decoder on top of frozen semantic features (Guo et al., 2026). At the other extreme, pixel-space models remove the reconstruction bottleneck and the cost of training a separate tokenizer. Clean-data prediction and large spatial or spatiotemporal patches make direct RGB diffusion practical without requiring an excessively

wide denoiser (Li & He, 2025; Jiang et al., 2026). Minimalist text-to-image models further demonstrate competitive generation without a learned visual tokenizer (Wang et al., 2026c), while unified generation–understanding models use pixel prediction to share representations across visual tasks (SenseNova-U1 Team, 2026).

Predictive representations for robot control. Pretrained visual representations such as R3M and V-JEPA 2-AC support data-efficient manipulation and latent planning (Nair et al., 2022; Assran et al., 2025). Other methods couple future prediction directly with policies, passing future information to control through generated frames (UniPi), jointly predicted images and actions (GR-1, GR-2), or intermediate video-diffusion features (VPP) (Du et al., 2023; Wu et al., 2023; Cheang et al., 2024; Hu et al., 2024). Fast-WAM finds that joint generation, inverse dynamics, and a variant without test-time imagination perform within about one point of each other, attributing most of the benefit to video co-training (Yuan et al., 2026).

Recent WAMs improve robustness by adding predictive priors: semantic foresight (Robust-WAM, Flex-π), multi-teacher distillation (GE-Act 2.0), V-JEPA feature prediction (CSWAM), or semantic and latent-dynamics objectives on top of reconstruction latents (ST-WAM, LaWAM) (Yan et al., 2026b;a; AgiBot Research Team, 2026; Liu et al., 2026; Wang et al., 2026b; Chen et al., 2026).

Comparing latent spaces for robot learning. OpenWAM trains one backbone on reconstruction latents and on DINOv3 or V-JEPA 2.1 features, and finds on in-domain RoboTwin that compact, information-rich latents matter more than the reconstructive–semantic divide (Wang et al., 2026d). Nilaksh et al. (2026) find that V-JEPA 2.1 yields the strongest policies among encoders for action-conditioned world models on BridgeV2, and concurrent work (ARPS) reports that a V-JEPA 2 teacher transfers far better to LIBERO-Plus than a DINOv2 teacher (Gu et al., 2026).

3 METHOD

Following the component view of WAMs (Lu et al., 2026), we decompose the policy into language-and-state conditioning, a visual codec, a predictive video backbone (the trunk), and an action expert. Video and action streams are denoised within one model, with their interaction specified by the attention mask and denoising schedule. This factorization lets us compare prediction spaces while preserving the trunk and action-expert architectures.

Figure 2 summarizes the framework. The representation variants share the video–action architecture and control pathway. Their encoders, channel adapters, and flow parameterization differ as summarized in Appendix Table 2. RAE-WAM uses either DINOv2 or V-JEPA features as its semantic representation. We use no representation-specific action head or auxiliary semantic loss.

Setup. A training sample is $(\ell, o_{1:F}, a_{1:T}, q, m)$, comprising a language instruction, video window, action chunk, proprioceptive state, and per-dimension validity mask. The frozen or parameter-free codec $\mathcal{E}$ maps $o_{1:F}$ to $1+K$ temporal latent slots z. The first slot is a clean anchor for the current observation, and (ℓ, q) forms the context c. Each stream is noised to its own timestep,

$$z^{t_v} = t_v\, z + (1-t_v)\,\epsilon_v, \qquad a^{t_a} = t_a\, a + (1-t_a)\,\epsilon_a, \qquad \epsilon_v, \epsilon_a \sim \mathcal{N}(0, I), \tag{1}$$

and one joint forward pass predicts both velocities,

$$(\hat{v}_z, \hat{v}_a) = v_\Theta\big(z^{t_v},\, a^{t_a},\, t_v,\, t_a,\, c\big), \qquad \Theta = (\theta, \phi), \tag{2}$$

where θ denotes the trunk and ϕ the action expert, including the projections that connect the two streams.

Representation codec. The codec is the primary controlled variable. Each frozen or parameter-free encoder produces $1+K$ slots, with the first slot anchoring the present,

$$\mathcal{E}:\ \mathbb{R}^{F\times 3\times H\times W} \longrightarrow \mathbb{R}^{(1+K)\times C\times h\times w}, \qquad z = \mathcal{E}(o_{1:F}), \tag{3}$$

and is patchified to a common per-slot token budget. Detailed codec and optimization configurations are reported in Appendix Table 2.

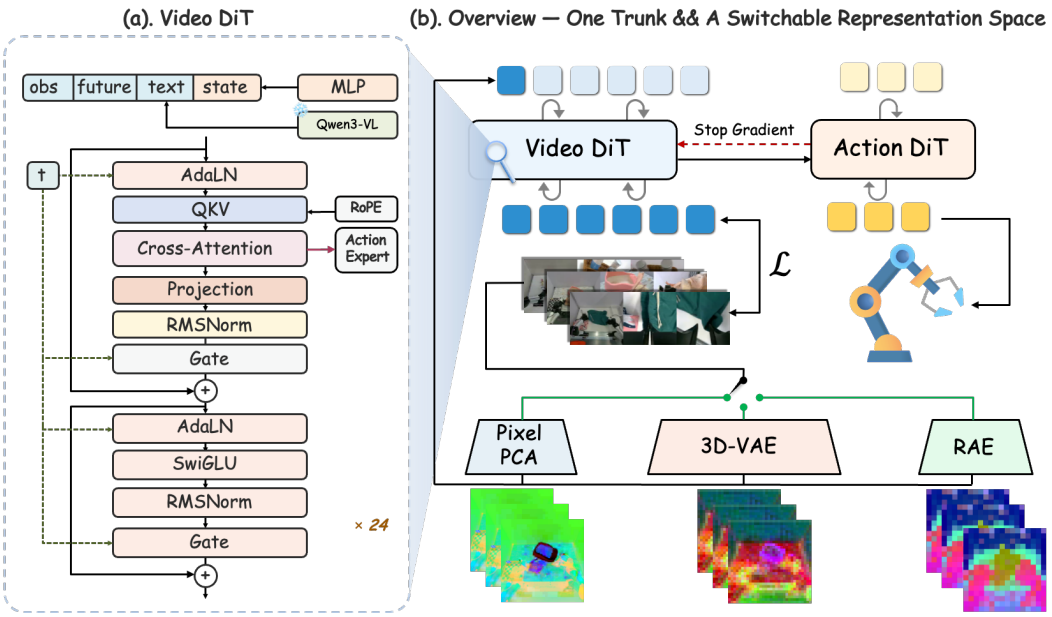

Figure 2: **Controlled world-action framework.** Left: the Video DiT block and the action expert. Right: the shared architecture with switchable visual codecs for Pixel-WAM, VAE-WAM, and RAE-WAM. RAE-WAM is instantiated with DINOv2 or V-JEPA 2.1 features.

We match the token budget and hold the trunk architecture, attention pattern, optimizer, and evaluation protocol fixed. VAE-WAM and both RAE-WAM variants share the same velocity-prediction flow recipe; Pixel-WAM uses a clean-sample readout with corresponding timestep and loss scaling. Appendix Table 2 makes these representation-specific differences explicit.

PIXEL (PIXEL-WAM). $\mathcal{E}$ directly packs normalized RGB frames into spatiotemporal patches, without a learned tokenizer. Repeating the first frame forms the conditioning slot.

VAE (VAE-WAM). $\mathcal{E}$ is the frozen Wan2.1 causal 3D VAE. Its first causal latent provides the conditioning slot.

SEMANTIC REPRESENTATION (RAE-WAM). RAE-WAM has two frozen encoder variants. *DINOv2* encodes frames independently and pools their image-pretrained features into future slots, serving as the static-semantic variant. *V-JEPA* jointly encodes the video window and temporally aggregates video-predictive features. Both use the first observation to form the clean conditioning slot.

For the Q2 representation comparison, every arm uses MoT-joint fusion. The Action DiT is randomly initialized once with a fixed seed, and the resulting parameter state is copied to every arm. Codec-dependent input and output projections remain in the video branch and are initialized separately, so the action-expert dimensionality and initialization are unchanged across codecs. The comparison therefore controls the action pathway, but characterizes complete representation variants rather than changing the encoder alone.

At inference, a *synchronized denoising schedule* means that the video and action ODEs use the same numerical integration grid. This does not require equal training noise levels: during training, t_v and t_a are sampled independently as defined below. The fusion design is specified for each experiment because Q1 ablates it, whereas Q2 fixes it to MoT-joint fusion.

Objective. We train the video and action streams with rectified-flow regression. Their timesteps are sampled independently, $t_v \perp\!\!\!\perp t_a$:

$$\mathcal{L} = \lambda_v \, \mathbb{E}_{t_v,\epsilon_v}\Big[w(t_v)\big\|\hat{v}_z - (z-\epsilon_v)\big\|_2^2\Big] + \lambda_a \, \mathbb{E}_{t_a,\epsilon_a}\Big[w(t_a)\big\|m \odot \big(\hat{v}_a - (a-\epsilon_a)\big)\big\|_2^2\Big], \quad (4)$$

where the conditioning slot is excluded from the video term and m restricts the action term to populated coordinates. Both streams use the same functional form $w(\cdot)$ at their independently sampled timesteps. Representation-specific readouts, timestep warps, loss weights, and target scaling are given in Appendix Table 2.

This formulation separates three choices that WAMs often entangle: the information preserved by the visual representation, the mechanism by which video and action exchange information (the fusion design), and the numerical schedule used to learn and integrate the flow field. Q1 studies the second choice. Q2 fixes the fusion design and action pathway while comparing complete representation variants; the three latent variants also share the flow recipe, whereas Pixel-WAM requires the pixel-specific recipe in Appendix Table 2.

4 Experiments

We treat closed-loop task success as the primary measure of a representation's utility for control. Generative loss and reconstruction quality are useful diagnostics, but neither substitutes for policy evaluation: perceptual fidelity, action accuracy, and closed-loop success measure different properties, and progress on one need not transfer to the others (Lu et al., 2026). We use *perceptual fidelity* for PSNR, SSIM, and LPIPS and distinguish it from preservation of control-relevant state such as object and end-effector position. For distribution-shift evaluation, we follow LIBERO-Plus and report controlled perturbations of background, camera viewpoint, language instruction, object layout, lighting, sensor noise, and robot initial state (Fei et al., 2025). Q1 and Q2 use independent pretraining setups. Q1 initializes the VAE-WAM trunk from LingBot-Video solely to compare fusion designs (Ma et al., 2026). Q2 instead uses xDoF pretraining on ABC-130K solely to pretrain the V-JEPA RAE-WAM; ABC-130K is an open-source teleoperation corpus containing approximately 3,500 hours and more than 130,000 episodes across 195 manipulation tasks (Allshire et al., 2026). We organize the experiments around three research questions:

- **Q1:** Through which fusion design do video-pretrained representations transfer most effectively to action prediction?
- **Q2:** Which representation space—pixels, VAE latents, or semantic representations—best supports closed-loop control, and how much does video pretraining help the selected representation?
- **Q3:** Does the LIBERO-Plus representation ordering persist in simulated bimanual manipulation, and do the gains of the selected model carry over to a real robot?

Within each ablation, we control the action-expert architecture, clip geometry, data, optimizer, and evaluation protocol. We vary the fusion design only in Q1 and fix it to MoT-joint for the LIBERO-Plus representation comparison. Q2 uses the shared Action DiT initialization specified in Section 3; the paired tests compare representation variants under this common control stack, with the codec interfaces and Pixel-WAM flow recipe treated as part of each arm.

Evaluation uncertainty. LIBERO-Plus evaluates one episode on each of 1,204 paired tasks. Values reported as mean $\pm$ uncertainty use a binomial 95% interval, $1.96\sqrt{p(1-p)/1204}$, and do not reflect variation across training seeds. Re-evaluating the same weights yields a per-task outcome-flip rate of 14% and an empirical standard deviation of 1.08 percentage points for paired aggregate differences. Shared-path timing, which excludes the representation encoders and is therefore not an end-to-end latency comparison, is reported in Section 4.3.

4.1 Q1: Transfer from video pretraining to action prediction

We first ask whether, and through which fusion design, video pretraining improves action prediction. This question must precede the codec comparison because the fusion design determines where visual

Table 1: **Fusion-design ablation.** Closed-loop success rates (%) on LIBERO-Plus overall and in its seven perturbation categories. The top panel initializes the VAE-WAM trunk from LingBot-Video pretraining; the bottom panel initializes the same trunk randomly. This experiment compares fusion designs within VAE-WAM. Overall is the mean over 1,204 tasks and $\pm$ is the binomial 95% interval for one episode per task. **Bold** and underlined entries denote the best and second-best results within each panel.

Fusion	Overall	Background	Camera	Language	Layout	Light	Noise	Robot
Pretrained trunk								
MoT (severed)	64.04 ± 2.71	45.3	57.0	58.3	81.6	**85.9**	70.5	56.0
Unified	68.36 ± 2.63	56.9	52.0	88.5	80.0	81.5	52.5	72.0
MoT (joint)	69.02 ± 2.61	51.8	61.0	86.5	81.1	79.3	53.5	71.5
MoE	71.35 ± 2.55	57.7	52.5	**93.2**	84.2	79.3	52.0	82.4
IDM	74.09 ± 2.48	66.4	53.5	90.6	86.8	77.2	54.5	**90.7**
Hidden	$\mathbf{81.06 \pm 2.21}$	**67.9**	**71.5**	**93.2**	**87.4**	83.7	**75.0**	87.0
Randomly initialized trunk								
MoT (joint)	45.43 ± 2.81	16.8	30.5	60.9	70.0	40.2	39.5	50.3
MoE	46.43 ± 2.82	10.9	27.0	65.6	73.7	26.1	47.0	54.9
MoT (severed)	48.92 ± 2.82	**19.7**	**48.0**	37.5	70.5	**59.8**	46.0	58.5
IDM	50.08 ± 2.82	10.9	19.0	74.5	79.5	32.6	44.0	71.5
Unified	51.58 ± 2.82	16.8	23.0	76.0	75.8	31.5	55.0	63.7
Hidden	$\mathbf{58.47 \pm 2.78}$	9.5	43.0	**78.1**	**84.2**	27.2	**63.0**	**74.6**

and action information interact, so a poor design can hide the value of the underlying representation (Lu et al., 2026). We compare six designs: hidden-state fusion, inverse-dynamics modeling (IDM), mixture-of-experts (MoE), joint and severed mixture-of-transformers (MoT), and unified token modeling (SenseNova-U1 Team, 2026). In hidden-state fusion, the action expert consumes video-trunk hidden states through a stop-gradient interface: video information flows forward to the action stream, but gradients from the action loss do not propagate back into the video trunk. This preserves access to pretrained video features while preventing the video and action objectives from imposing conflicting updates on the trunk. For the pretrained condition, we initialize the VAE-WAM trunk from the LingBot-Video video-pretraining checkpoint (Ma et al., 2026). We compare every fusion design against its randomly initialized counterpart, isolating the extent to which each design can exploit that fixed visual initialization.

Video pretraining transfers most effectively through hidden states. Table 1 shows that hidden-state fusion is the strongest and most consistent design under the LingBot-Video VAE initialization. It reaches 81.06% overall success, 6.97 points above the next-best design, IDM, and is best or tied for best on five of seven perturbation categories. The advantage over IDM is concentrated in camera and noise shifts, which together account for 91% of the gap, and both differences remain significant after Holm correction. IDM performs best under robot initial-state shifts (90.7%), while severed MoT leads under lighting changes but degrades sharply on background, language, and robot initial-state shifts. Hidden-state fusion also ranks first from scratch at 58.47%, exceeding unified token modeling by 6.89 points. LingBot-Video initialization improves every fusion design by 15.12–24.92 points. Its stop-gradient interface provides a plausible explanation for the strong transfer: the action expert can use pretrained video hidden states without allowing action gradients to overwrite the video dynamics prior. These results concern fusion transfer within the Q1 VAE-WAM setup only. Figure 3 provides a layer-wise diagnostic of this behavior. Across the six pretrained fusion designs, the share of action-token attention assigned to future slots is positively associated with closed-loop success ($r_s = 0.83$, exact two-sided $p = 0.058$), although we treat this small-sample result as descriptive. In hidden-state fusion, 68% of action-expert attention falls on future slots. Video pretraining reduces the text-attention share in five of the six matched pairs, while zeroing text increases action MSE by 55–337$\times$ for MoT-family variants but by 2.6$\times$ for hidden-state fusion. Open-loop metrics do not explain the closed-loop ranking: hidden-state fusion has the worst action MSE among the six designs—7.6$\times$ that of severed MoT—while attaining the highest video PSNR (23.36 dB) and the best closed-loop success.

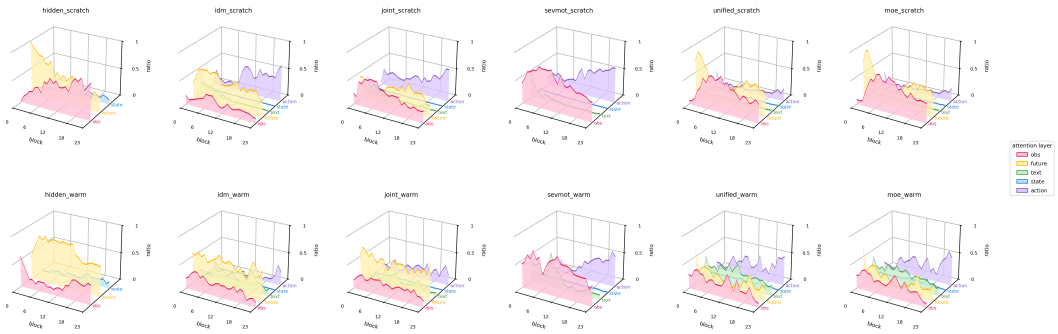

Figure 3: **Layer-wise modality attention across fusion designs.** Attention ratios assigned to observation, future, text, state, and action tokens across transformer blocks for hidden-state fusion, IDM, joint MoT, severed MoT, unified token modeling, and MoE. The top row shows randomly initialized trunks and the bottom row shows video-pretrained trunks.

4.2 Q2: Representation spaces and video pretraining

We next compare the three representation-space families named in the title: Pixel-WAM, VAE-WAM, and RAE-WAM. All four variants aggregate four future frames per slot, so temporal compression itself does not distinguish them. Within RAE-WAM, DINOv2 averages frame-wise image-pretrained features, whereas V-JEPA 2.1 learns video-predictive temporal features from the joint video window. The comparison therefore tests whether the learned structure of the prediction space—not merely its compression ratio or semantic abstraction—matters for control. All variants predict the same future-video target in their respective spaces and expose the resulting representation to the action expert. The Pixel-WAM+bottleneck variant changes only the input patch embedding to a low-rank linear factorization, $\mathrm{Linear}(3072, 512, \mathrm{bias{=}false})$ followed by $\mathrm{Linear}(512, 2048)$, with no intervening nonlinearity or convolution and with the output projection unchanged. It tests whether the nominal pixel-patch dimension is redundant, rather than serving as a general channel-capacity control.

All LIBERO-Plus runs in this comparison use MoT-joint fusion and the common Action DiT initialization described in Section 3. The trunk architecture, action pathway, data, optimizer, temporal window, slot count, token budget, and evaluation tasks are shared. VAE-WAM and both RAE-WAM variants additionally share the velocity-prediction flow recipe; Pixel-WAM instead uses the clean-sample readout, timestep sampling, and loss scaling listed in Appendix Table 2. Results therefore compare complete representation variants under a shared control stack rather than attributing every difference to the encoder alone. Q3 separately uses the best-performing Q1 design, hidden-state fusion, for the available Pixel-WAM, VAE-WAM, and V-JEPA RAE-WAM models on RoboTwin.

Video-predictive semantic features provide the strongest generalization. Figure 4(a) shows that, under MoT-joint fusion, the V-JEPA RAE-WAM achieves the highest aggregate success at 55.15%, exceeding the DINOv2 RAE-WAM by 9.47 points and VAE-WAM by 9.72 points. Paired McNemar tests over the 1,204 tasks confirm both differences: the V-JEPA RAE-WAM versus VAE-WAM has 216/99 discordant wins ($p = 3.8 \times 10^{-11}$), and the two RAE-WAM variants have 272/158 ($p = 4.2 \times 10^{-8}$). The DINOv2 RAE-WAM and VAE-WAM are tied in aggregate (213/210, $p = 0.92$), so the evidence does not support a class-level claim that all semantic representations beat reconstruction latents. All three latent variants significantly outperform Pixel-WAM ($p < 10^{-5}$). The low-rank embedding of Pixel-WAM+bottleneck adds only 1.66 points (145/125, $p = 0.25$), providing no significant evidence that nominal pixel-patch dimensionality explains the gap.

The factor profiles nevertheless differ. The DINOv2 RAE-WAM leads under background and lighting changes, whereas the V-JEPA RAE-WAM leads under camera, language, layout, and robot initial-state shifts. In a descriptive codec probe, adding future slots beyond a first-frame-only control contributes incremental video-to-action R^2 of 0.21 for the V-JEPA RAE-WAM, 0.20 for VAE-WAM, 0.16 for Pixel-WAM, and 0.13 for the DINOv2 RAE-WAM. This ordering is only partially aligned with closed-loop success: V-JEPA and VAE have nearly identical probe values despite a

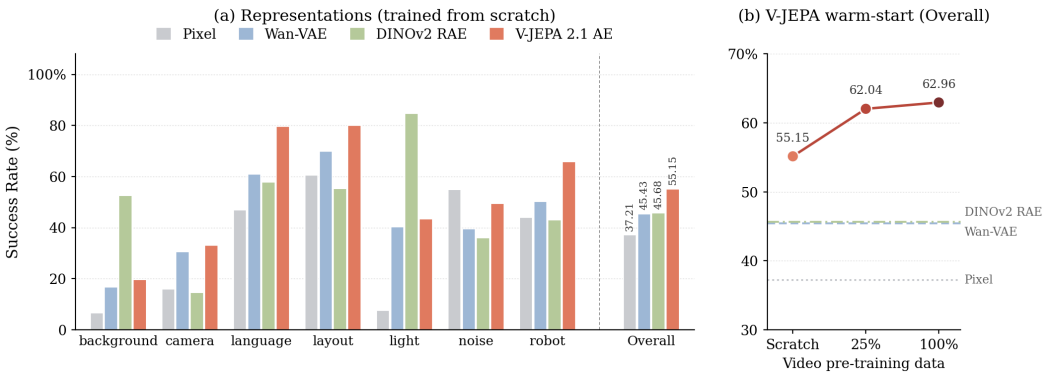

Figure 4: **Representation choice and xDoF video pretraining.** (a) Closed-loop success with MoT-joint fusion, a randomly initialized trunk, and the shared random Action DiT initialization described in Section 3 for Pixel-WAM, VAE-WAM, and the DINOv2 and V-JEPA 2.1 variants of RAE-WAM. (b) V-JEPA RAE-WAM success from scratch and after xDoF pretraining on the full ABC-130K corpus. This pretraining is specific to V-JEPA RAE-WAM. Dashed horizontal lines show the scratch baselines of the other variants.

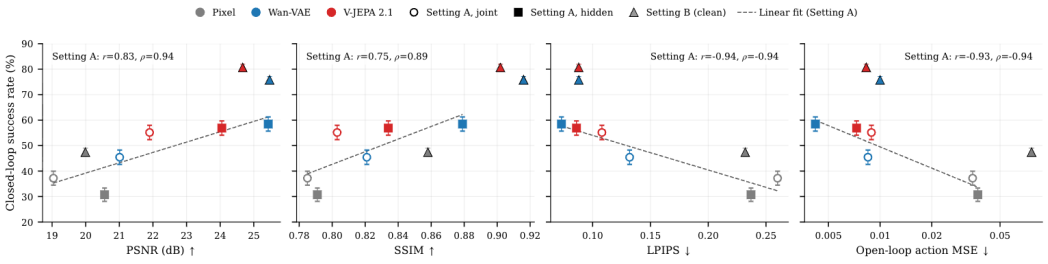

Figure 5: **Open-loop metrics versus closed-loop success.** Closed-loop success is plotted against PSNR, SSIM, LPIPS, and open-loop action MSE for Pixel-WAM, VAE-WAM, and the V-JEPA RAE-WAM. Setting A denotes LIBERO-Plus: open circles are MoT-joint models and filled squares are hidden-state-fusion models. Setting B denotes clean RoboTwin evaluations, shown as triangles. Dashed lines and the reported Pearson r and Spearman ρ values are fitted only to Setting A.

9.72-point success gap, while Pixel exceeds DINOv2 in the probe but performs worse in control. The probe is therefore consistent with V-JEPA retaining future-dependent information, but it neither explains the performance gaps nor establishes a causal mechanism. The shared fusion and action expert reduce control-side confounds; remaining differences include the representation codecs and interfaces, and Pixel-WAM's distinct flow recipe.

Video pretraining yields a clear gain. Figure 4(b) compares the V-JEPA RAE-WAM trained from scratch with the model pretrained through xDoF pretraining on the full ABC-130K corpus. This pretraining raises closed-loop success from 55.15% to 62.96% ($p < 10^{-6}$), a gain of 7.81 percentage points within V-JEPA RAE-WAM.

Figure 5 compares closed-loop success with perceptual fidelity and open-loop action MSE. We call the LIBERO-Plus MoT-joint and hidden-state-fusion models *Setting A*, and the clean RoboTwin evaluations *Setting B*. The LingBot-Video initialization in Q1 is used only for the VAE-WAM fusion ablation, whereas ABC-130K in Q2 is used only to pretrain V-JEPA RAE-WAM. Their results belong to independent experiments and do not form a cross-representation pretraining comparison. Figure 4(b) therefore establishes only the within-V-JEPA effect of ABC-130K pretraining.

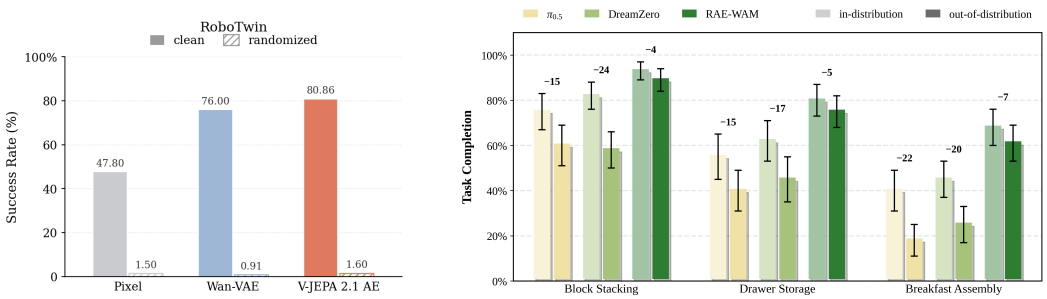

Figure 6: **Bimanual manipulation evaluation.** Left: RoboTwin success rates for Pixel-WAM, VAE-WAM, and the V-JEPA RAE-WAM in clean and visually randomized environments. Right: ID and OOD real-robot completion rates of $\pi_{0.5}$, DreamZero, and the V-JEPA RAE-WAM across three tasks.

4.3 Q3: Transfer to bimanual manipulation in simulation and reality

We next evaluate bimanual manipulation in RoboTwin 2.0 simulation and on a physical dual-arm platform. The simulation study tests whether the representation ordering persists under the best Q1 fusion design. The real-robot study instead compares the selected V-JEPA RAE-WAM with strong policy baselines and therefore does not support a representation-ordering claim.

RoboTwin setup. RoboTwin 2.0 evaluates bimanual manipulation under structured visual randomization (Chen et al., 2025). Unlike the MoT-joint LIBERO-Plus representation comparison, Pixel-WAM, VAE-WAM, and the V-JEPA RAE-WAM use hidden-state fusion here. The DINOv2 RAE-WAM was not evaluated.

The simulation ordering transfers, but randomization remains difficult. Figure 6 reports clean/randomized success of $47.80/1.50\%$ for Pixel-WAM, $76.00/0.91\%$ for VAE-WAM, and $80.86/1.60\%$ for the V-JEPA RAE-WAM. Under the same hidden-state fusion design, the clean ordering matches LIBERO-Plus: the V-JEPA RAE-WAM performs best, followed by VAE-WAM and then Pixel-WAM. All three models collapse under severe visual randomization. The V-JEPA RAE-WAM improves the clean starting point but does not solve this domain shift.

Real-robot results. On the dual-Franka platform shown in Appendix Figures 7 and 8, the V-JEPA RAE-WAM outperforms $\pi_{0.5}$ (Physical Intelligence et al., 2025) and DreamZero (Ye et al., 2026) in both ID and OOD evaluation on all three tasks. Its ID/OOD completion rates are $94/90\%$ on block stacking, $81/76\%$ on drawer storage, and $69/62\%$ on breakfast assembly. Deployment timing is reported in the appendix.

5 Conclusion

We compared prediction spaces and fusion designs for world-action models. In Q1, hidden-state fusion transfers the LingBot-Video VAE initialization more effectively than the other five designs; its stop-gradient interface likely protects the video prior from action-loss interference. Building on this, V-JEPA RAE-WAM achieves the best overall success, while DINOv2 RAE-WAM and VAE-WAM are tied. Because every arm aggregates four future frames per slot, this result indicates that video-predictive temporal structure, rather than temporal compression alone, is the key factor. The codec probe supports the view that V-JEPA retains future-dependent information, while also showing that closed-loop success is not explained by simple information measures. ABC-130K pretraining further improves V-JEPA RAE-WAM over its scratch baseline. The same ranking appears in clean RoboTwin, and on the real dual-arm platform V-JEPA RAE-WAM outperforms $\pi_{0.5}$ and DreamZero. These results suggest a practical design principle for world-action models: predict the future in a pretrained semantic space whose temporal structure is learned through video-predictive training, rather than in pixels or reconstruction-oriented VAE latents.

AI USE STATEMENT

Large language models were used to assist with programming and figure preparation, as well as grammar and expression polishing. They did not contribute to research ideation, methodology design, or experiments.

ETHICS STATEMENT

This work complies with the ICLR Code of Ethics. It does not involve human or animal subjects, nor the use of private or sensitive data. The research raises no direct ethical or legal concerns, and the authors are committed to responsible and fair use of the proposed methods.

REPRODUCIBILITY STATEMENT

We have made every effort to ensure the reproducibility of our work. The proposed model, implementation details, and evaluation protocols are described in detail in the main paper and appendix. The source code and data will be made publicly available.

REFERENCES

AgiBot Research Team. GE-Act 2.0: Pretraining and scaling a world-action model for robotic manipulation. *arXiv preprint arXiv:2609.05588*, 2026.

Arthur Allshire, Himanshu Gaurav Singh, Ritvik Singh, Adam Rashid, Hongsuk Choi, David McAllister, Justin Yu, Yiyuan Chen, Huang Huang, Pieter Abbeel, Xi Chen, Rocky Duan, Phillip Isola, Jitendra Malik, Fred Shentu, Guanya Shi, Philipp Wu, and Angjoo Kanazawa. Scalable behavior cloning with open data, training, and evaluation. *arXiv preprint arXiv:2606.27375*, 2026.

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-JEPA 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

Chi-Lam Cheang, Guangzeng Chen, Ya Jing, Tao Kong, Hang Li, Yifeng Li, Yuxiao Liu, Hongtao Wu, Jiafeng Xu, Yichu Yang, Hanbo Zhang, and Minzhao Zhu. GR-2: A generative video-language-action model with web-scale knowledge for robot manipulation. *arXiv preprint arXiv:2410.06158*, 2024.

Jialei Chen, Kai Wang, Kang Chen, Shuaihang Chen, Feng Gao, Wenhao Tang, Zhiyuan Li, Weilin Liu, Zhuyu Yao, Boxun Li, Yuanbo Xu, and Chao Yu. LaWAM: Latent world action models for efficient dynamics-aware robot policies. *arXiv preprint arXiv:2606.15768*, 2026.

Tianxing Chen, Zanxin Chen, Baijun Chen, Zijian Cai, Yibin Liu, Zixuan Li, Qiwei Liang, Xianliang Lin, Yiheng Ge, Zhenyu Gu, Weiliang Deng, Yubin Guo, Tian Nian, Xuanbing Xie, Qiangyu Chen, Kailun Su, Tianling Xu, Guodong Liu, Mengkang Hu, Huan-ang Gao, Kaixuan Wang, Zhixuan Liang, Yusen Qin, Xiaokang Yang, Ping Luo, and Yao Mu. RoboTwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation. *arXiv preprint arXiv:2506.18088*, 2025.

Yilun Du, Mengjiao Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Joshua B. Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. *arXiv preprint arXiv:2302.00111*, 2023.

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.

Qiwen Gu, Jifan Li, Bingjie Gao, Rui Chen, Jing Tang, Xiangxiang Chu, and Junqiao Zhao. The right future for action: Learning action-relevant predictive states in world action models. *arXiv preprint arXiv:2609.23369*, 2026.

Minghui Guo, Shengqiong Wu, and Hao Fei. V-RAE: Rethinking video latent spaces for generation. *arXiv preprint arXiv:2608.13556*, 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.

Dengyang Jiang, Ruoyi Du, Zhennan Chen, Dongyang Liu, Zanyi Wang, Mingzhe Zheng, Xiangpeng Yang, Huanqia Cai, Aiming Hao, Yuming Jiang, Peng Gao, Harry Yang, and Steven Hoi. An empirical study of training pixel-space text-to-image diffusion models. *arXiv preprint arXiv:2608.16887*, 2026.

Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise. *arXiv preprint arXiv:2511.13720*, 2025.

Tianbin Liu, Jian Zhu, Taiyi Su, Jianjun Zhang, Chong Ma, Zitai Huang, Weiyi Lu, and Yi Xu. CSWAM: Better causal semantic representations for out-of-distribution generalization in world action models. *arXiv preprint arXiv:2609.18462*, 2026.

Zuxing Lu, Hongjia Zhai, Guanzhi Wang, Huajian Zeng, Jiaqi Yang, Jingyu Liu, Lei Cheng, Yuantai Zhang, Yuheng Qiu, Zezhou Cheng, Ivan Laptev, Danfei Xu, Benjamin Riviere, Giuseppe Loianno, Eric Xing, and Xingxing Zuo. World-action models for robot learning and control: A survey. *arXiv preprint arXiv:2609.16074*, 2026.

Shuailei Ma, Jiaqi Liao, Xinyang Wang, Jingjing Wang, Chaoran Feng, Zijing Hu, Chong Bao, Zichen Xi, Yuqi Gan, Weisen Wang, Yanhong Zeng, Qin Zhao, Zifan Shi, Wei Wu, Hao Ouyang, Qiuyu Wang, Shangzhan Zhang, Jiahao Shao, Yipengjing Sun, Liangxiao Hu, Lunke Pan, Nan Xue, Kecheng Zheng, Yinghao Xu, Xing Zhu, Yujun Shen, and Ka Leong Cheng. Scaling mixture-of-experts video pretraining for embodied intelligence. *arXiv preprint arXiv:2607.07675*, 2026.

Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3M: A universal visual representation for robot manipulation. *arXiv preprint arXiv:2203.12601*, 2022.

Nilaksh, Saurav Jha, Artem Zholus, and Sarath Chandar. Reconstruction or semantics? what makes a latent space useful for robotic world models. *arXiv preprint arXiv:2605.06388*, 2026.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: A vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025.

SenseNova-U1 Team. SenseNova-U1: Unifying multimodal understanding and generation with NEO-unify architecture. *arXiv preprint arXiv:2605.12500*, 2026.

Letian Wang, Chuhan Zhang, Rishabh Kabra, Jasper Uijlings, Steven Waslander, Andrew Zisserman, Joao Carreira, Kaiming He, Misha Andriluka, Eduard Gabriel Bazavan, Andrei Zanfir, and Cristian Sminchisescu. Video generation models are general-purpose vision learners. *arXiv preprint arXiv:2607.09024*, 2026a.

Mingxin Wang, Bin Hu, Bin Qian, Kaitao Jiang, Haoning Wu, Feng Yan, Bowen Jing, Ruiyang Hao, Enyi Wang, Kangning Niu, Yandan Yang, Mu Xu, Yan Wang, Houde Liu, and Tianlun Li. ST-WAM: Semantic-temporal world action model for robust manipulation under visual distribution shifts. *arXiv preprint arXiv:2607.28993*, 2026b.

Xianbang Wang, Hanhong Zhao, Yiyang Lu, Kangyang Zhou, Linrui Ma, and Kaiming He. MiniT2I: A minimalist baseline for text-to-image generation, 2026c. URL https://peppaking8.github.io/#/post/minit2i.

Yuran Wang, Siqiao Huang, Mingleyang Li, Chenhao Zhang, Jiaqi Liang, Weiyang Jin, Yue Chen, Xuemin Chi, Donghao Zhou, Qize Yu, Yu-Kai Wang, Yuhan Rui, Shenzhe Yao, Zhen Yuan, Zhenhao Shen, Kefei Zhu, Zijie Zhu, Ning Gao, Xiaowei Chi, Guanqi He, Shanghang Zhang, Hao Dong, Lin Shao, and Hang Zhao. OpenWAM: An open, modular exploration towards systematic world-action model pretraining. *arXiv preprint arXiv:2609.07398*, 2026d.

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

Ge Yan, Jinghao Liu, Yuzhi Fan, Lei Cai, Minwen Liao, Jesse Zhang, and Dieter Fox. Flex-π: A multi-stream world-action model with compute flexibility. *arXiv preprint arXiv:2608.10860*, 2026a.

Haodong Yan, Junfeng Li, Junjie He, Zhide Zhong, MingMing Yu, Wenxuan Song, Jiaguan Zhu, Yangyang Zheng, Yuqiao Du, Jiadi You, Yingjie Cai, Xu Yan, Guanyi Zhao, Bingbing Liu, and Haoang Li. Robust-WAM: Bridging generative pretraining and semantic foresight in world-action models. *arXiv preprint arXiv:2608.05903*, 2026b.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, Ayaan Malik, Kyungmin Lee, William Liang, Nadun Ranawaka, Jiasheng Gu, Yinzhen Xu, Guanzhi Wang, Fengyuan Hu, Avnish Narayan, Johan Bjorck, Jing Wang, Gwanghyun Kim, Dantong Niu, Ruijie Zheng, Yuqi Xie, Jimmy Wu, Qi Wang, Ryan Julian, Danfei Xu, Yilun Du, Yevgen Chebotar, Scott Reed, Jan Kautz, Yuke Zhu, Linxi Jim Fan, and Joel Jang. World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*, 2026.

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

Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders. *arXiv preprint arXiv:2510.11690*, 2025.

Tianxiong Zhong, Xingye Tian, Xuebo Wang, Xin Tao, and Pengfei Wan. Diffusing in the right space: A systematic study of latent diffusability. *arXiv preprint arXiv:2606.03578*, 2026.

A Detailed Method Configuration

Table 2: **Representation and optimization configurations.** All variants predict $K = 2$ future slots with $N = 16 \times 32 = 512$ tokens per slot and use action-loss weight $\lambda_a = 1$. The three latent variants share the v-prediction recipe, whereas Pixel-WAM uses an x-prediction recipe with different timestep and loss scaling. Here p is the effective temporal–spatial patch or tubelet size, ρ parameterizes the timestep warp, and s_x is the clean-target scale applied before noising.

Arm	Encoder / packing	C	p	K	N	Readout	ρ	$w(t_v)$	(λ_v, s_x)
Pixel-WAM	RGB packing	12	$(4, 16, 16)$	2	512	x-pred	1	logit-$\mathcal{N}(0.8, 0.8)$	$(64, 1/8)$
VAE-WAM	Wan2.1 causal 3D VAE	16	$(1, 2, 2)$	2	512	v-pred	5	Gaussian	$(1, 1)$
DINOv2 RAE-WAM	DINOv2-L/14	1024	$(1, 1, 1)$	2	512	v-pred	5	Gaussian	$(1, 1)$
V-JEPA RAE-WAM	V-JEPA 2.1	64	$(1, 1, 1)$	2	512	v-pred	5	Gaussian	$(1, 1)$

Temporal construction. Pixel-WAM packs four normalized RGB frames along the channel axis and repeats the first frame to construct the conditioning slot. VAE-WAM uses $4\times$ temporal and $8\times$ spatial compression, with the first causal latent as conditioning. Both semantic variants use the same nine-frame window. DINOv2 encodes frames independently, interpolates features to the 16×32 grid, and averages frames 2–5 and 6–9 into two future slots; the first-frame features form the conditioning slot. V-JEPA jointly encodes the window, aggregates each group of four future frames into one slot, and standardizes features with dataset statistics. Because V-JEPA is non-causal, its conditioning slot is obtained from a constant clip that repeats the first frame.

Flow conventions. For $\rho > 1$, the shifted timestep is $\tilde{t} = \rho t/[1 + (\rho - 1)t]$; "Gaussian" in Table 2 denotes Gaussian timestep weighting. A clean-sample prediction is converted to velocity as $\hat{v} = (\hat{x} - x^t)/\max(1 - t, \epsilon_t)$. Pixel-WAM divides its clean target by the noise scale 8 and uses $\lambda_v = 8^2 = 64$ to restore comparable units, so this coefficient performs dimensional alignment rather than introducing an independently tuned task weight. The conditioning slot is excluded from the video loss, and the action mask retains only populated coordinates.

B Real-Robot Setup and Tasks

Shared-path latency. On an H200 with three cameras, text encoding takes 25.8 ms, action-model prefill 33.5 ms, and each action-expert step 22.5 ms, for approximately 290 ms over ten denoising steps. These measurements exclude the representation-specific VAE, DINOv2, and V-JEPA encoder costs and therefore do not compare end-to-end deployment latency.

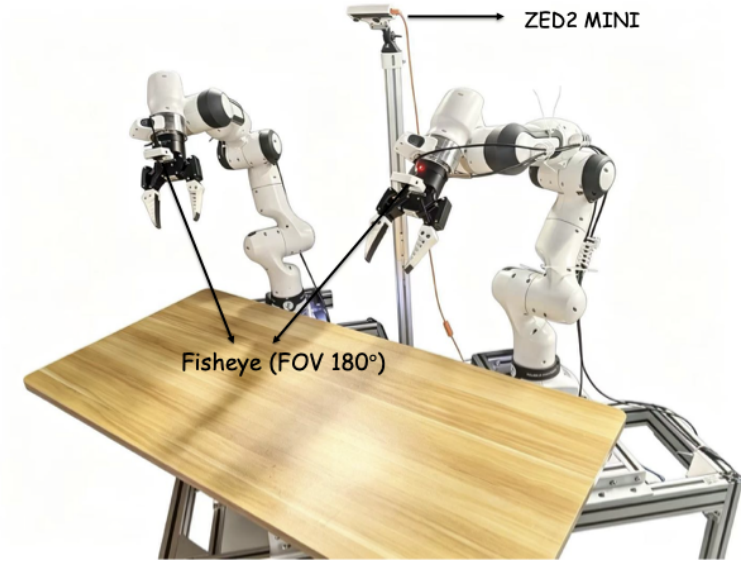

Figure 7: **Real-robot setup.** Dual-Franka tabletop platform with global and wrist-mounted cameras.

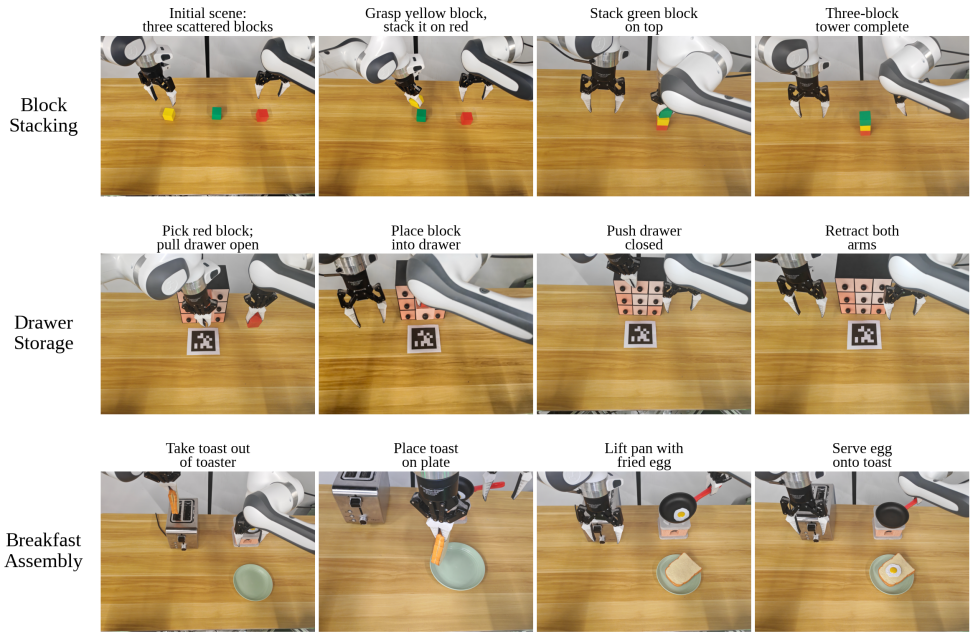

Figure 8: **Real-robot task suite.** Representative stages from three long-horizon bimanual tasks: stacking three colored blocks, storing a block in a drawer, and assembling a breakfast plate with toast and a fried egg.