JEPAGeneralist: Generalizing Action-Conditioned Latent World Models with Video Model Rewards

Haichao Zhang[1] **Yijiang Li**[2] **Shwai He**[3] **Van Thanh Le** [1]
Nuno Vasconcelos [2] **Yun Fu**[1]

[1]Northeastern University
[2]University of California San Diego
[3]University of Maryland, College Park

{zhang.haich,le.thanh1}@northeastern.edu, yunfu@ece.neu.edu
{yijiangli,nvasconcelos}@ucsd.edu, shwaihe@umd.edu,

Abstract

JEPA-style latent world models offer an efficient alternative to pixel-space video generation by predicting future representations rather than synthesizing future frames. However, current action-conditioned predictors in such latent world models are typically trained on single or narrowly scoped embodied datasets with limited task, object, embodiment, and action-space coverage, and often fail to generalize beyond the training distribution. We argue that this lack of generalist action-conditioned prediction is a key bottleneck for scaling latent world models. In contrast, large controllable video world models have a clear scale-up path through large-scale video pretraining and encode broad visual-dynamics priors, but they are too computationally expensive to serve as online simulators and lack a direct interface to latent action-conditioned prediction. We introduce JEPAGeneralist, a framework for transferring generalist motion priors from video world models to efficient action-conditioned JEPA predictors, establishing a scalable training path for future latent world models. Instead of directly distilling generated futures, whose latents entangle useful action dynamics with source-mismatched content and generation artifacts, JEPAGeneralist factorizes future latent prediction into source-conditioned content continuation and action-induced dynamics. Controllable video world models are used as same-action environment randomizers to build an offline motion-reward bank, providing diverse motion supervision without treating generated content as an absolute target. Specifically, generated rollouts are encoded with a frozen JEPA encoder, and only their patch-level temporal deltas are distilled as video-world-model motion rewards for predicted latent dynamics. This design preserves the efficiency of latent rollout, since video generation and representation encoding are performed offline, while leveraging the broader visual-dynamics priors of video world models. We evaluate JEPAGeneralist on three complementary benchmarks: EgoDex for held-out subtask generalization and action recoverability, BAIR robot pushing for future-frame latent prediction and action sensitivity, and Physion for physical contact readout. Our results show that video-world-model motion rewards improve latent prediction, action sensitivity, action recoverability, and physical contact readout beyond real-only training and naive generated-latent distillation, addressing a key generalization bottleneck on the path toward scaling up action-conditioned latent world models.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

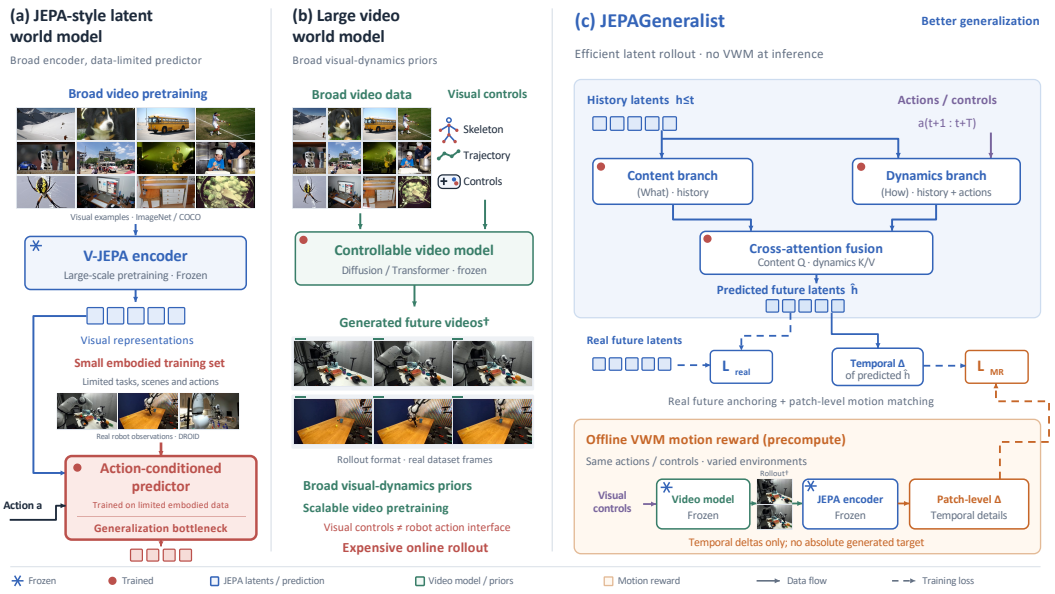


Figure 1: **Generalization mismatch in current latent world models.** JEPA-style systems often combine a broadly pretrained visual encoder with an action-conditioned predictor trained on much narrower embodied data. While the V-JEPA encoder benefits from large-scale video pretraining and can provide general visual representations, the action-conditioned predictor is typically trained or fine-tuned on limited action-labeled datasets, such as DROID or task-specific robot data, and therefore becomes the main generalization bottleneck. In contrast, large video world models are trained on broad video corpora and acquire more general visual-dynamics priors, but are too expensive to use directly for online rollout. JEPAGeneralist addresses this mismatch by using frozen video world models as offline sources of patch-level motion rewards from same-action rollouts, while keeping predictor training and inference in the efficient JEPA latent space.

1 Introduction

JEPA-style latent world models have emerged as an efficient alternative to pixel-space video world models. Rather than generating future frames through autoregressive decoding or iterative denoising, they predict future representations in a learned latent space [55, 9, 4, 106, 100, 81]. This representation-level prediction is attractive for robotics [25, 84, 56], embodied AI [64, 21, 60], autonomous driving [19, 44], computer vision [42, 28, 38, 75, 22, 61, 77, 58, 35], and representation learning [41, 23, 2], where fast rollout and scalable prediction are essential. In principle, an action-conditioned JEPA predictor can serve as a compact latent dynamics model: given past visual latents and a planned action or control sequence, it predicts the latent representation of the resulting future state without the cost of detailed pixel synthesis.

However, generalization has become a central bottleneck for action-conditioned JEPA-style latent world models. As illustrated in Figure 1, current approaches face a difficult trade-off. Vision-language-action systems often adapt pretrained vision-language models to action prediction [66, 15, 52], but task-specific fine-tuning on limited embodied data can weaken pretrained generalist capabilities and may suffer from catastrophic forgetting [53], leaving robust cross-task and cross-embodiment generalization challenging. In contrast, large controllable video world models scale through broad video pretraining and acquire diverse visual-dynamics priors [13, 17, 87, 59], but their pixel-space diffusion or flow-based rollouts are computationally expensive and high-latency. JEPA-style latent world models offer efficient rollout, but this efficiency cannot be fully leveraged without a generalist action-conditioned predictor.

This raises a natural question: can we retain the efficiency of latent world models while borrowing the broad visual-dynamics priors of video world models? Controllable video world models are trained on broad video corpora and can absorb diverse motion, interaction, and environment priors without requiring every trajectory to be paired with a precisely aligned robot action. This makes them complementary to action-conditioned latent world models, whose predictors are often trained on

embodied datasets with limited task, object, embodiment, and action-space coverage. We therefore study frozen video world models not as competitors to latent world models, but as scalable sources of generalist motion priors for training them.

Transferring these priors is non-trivial. First, the two model families do not share a natural interface: JEPA predictors operate in latent representation space, whereas controllable video world models are typically conditioned through pixel- or video-space controls such as VACE-style controls [50], skeleton maps, trajectory maps, or other spatial guidance. Action representations are also heterogeneous, ranging from hand poses and fingertips to end-effector trajectories, object centers, and robot joints. Second, naively distilling generated future latents can be harmful. Generated videos may preserve the intended action while changing the environment, object identity, hand appearance, camera layout, or generation style; after JEPA encoding, their latents entangle useful action dynamics with source-mismatched content and generation artifacts. Thus, generated futures should not be treated as absolute future targets. Instead, they should supervise the action-induced dynamics component while real future videos anchor the source-conditioned content continuation.

We present JEPAGeneralist, a framework for generalizing action-conditioned JEPA-style latent world models with video-world-model motion rewards. The key idea is to factorize future latent prediction into source-conditioned content continuation and action-induced dynamics. To obtain scalable motion supervision, we render action proxies or candidate control sequences into unified visual controls, use a frozen controllable video world model to generate same-action future rollouts, encode these rollouts with a frozen JEPA encoder, and cache them offline. Instead of directly distilling generated future latents, JEPAGeneralist distills only their patch-level temporal deltas as offline motion-reward supervision. This allows the predictor to exploit same-action rollouts across diverse environments while avoiding direct supervision from source-mismatched generated content. At inference time, JEPAGeneralist remains a lightweight action-conditioned latent predictor and requires no video world model, pixel-level generation, video decoding, or generated-video JEPA encoding.

We evaluate JEPAGeneralist on three complementary benchmarks. On EgoDex, we test held-out subtask generalization and measure whether predicted future latents preserve recoverable embodied action dynamics. On BAIR robot pushing, we evaluate future-frame latent prediction and action sensitivity in the V-JEPA latent space. On Physion, we evaluate whether predicted future latents preserve physically relevant contact information through a contact readout protocol. Together, these evaluations test whether the learned predictor improves latent prediction accuracy, action recoverability, and physical readout quality.

Contributions. (1) Generalist action-conditioned JEPA prediction. We study the generalization bottleneck of action-conditioned JEPA-style latent world models and formulate a held-out embodied-category evaluation setting, instantiated as held-out subtask generalization on EgoDex. This setting tests whether a single latent predictor can generalize beyond the action and task support observed during training; **(2) Content–dynamics factorized latent predictor.** We introduce a factorized action-conditioned JEPA predictor that separates source-conditioned content continuation from action-induced dynamics. In our main calibrated variant, instantiated with Cross-Attn fusion, a content-anchored cross-attention module injects motion cues while preserving the calibrated JEPA latent manifold; **(3) Offline video-world-model motion rewards.** We use frozen controllable video world models as same-action environment randomizers and build an offline motion-reward bank from generated future rollouts. Rather than treating generated futures as absolute latent targets, we encode them with a frozen JEPA encoder and distill only their patch-level temporal deltas as offline motion-reward supervision; **(4) Generalization, action-recoverability, and physical-readout evaluation.** We evaluate JEPAGeneralist on EgoDex, BAIR robot pushing, and Physion using latent prediction, action sensitivity, component ablations, IDM-based action recoverability, and contact readout metrics. The results show that JEPAGeneralist improves calibrated latent prediction, preserves more recoverable action dynamics, and produces future latents with stronger physically relevant contact information.

2 Related Works

2.1 World Models

JEPA-style latent world models. JEPA-style models learn predictive representations by forecasting latent states rather than reconstructing pixels, following the broader proposal that world models

should operate in abstract representation spaces [55]. This idea has been instantiated for images [3] and extended to videos for feature-level prediction, representation learning, and planning [9, 4, 30]. Latent world models further connect this idea to control by modeling dynamics over pretrained visual features, enabling zero-shot or fine-tuned planning from offline behavioral and observation-action trajectories [106, 7]. These works make latent rollout efficient, but they also highlight a remaining mismatch: broad pretraining mainly benefits the visual encoder, while the action-conditioned predictor is still learned from the available latent-action data. Its generalization is therefore limited by the tasks and embodiments used for training. Our work focuses on this bottleneck by generalizing the action-conditioned predictor itself.

Video world models and controllable generation. Early video diffusion models showed the promise of learning visual dynamics in pixel or latent video spaces [13, 12, 95, 79]. Recent open-weight models further scale this direction with efficient latent video generation and DiT/flow-matching-style training [40, 54, 87, 89, 96, 34], while proprietary systems such as Sora [17] and Veo [39] demonstrate strong general-purpose video generation. Recent controllable video models further expose different control interfaces: motion and trajectory controls [94, 91, 36, 83, 50], physics and force controls [88, 37], and action-interactive world controls [18, 85, 8]. However, using them directly for online rollout is costly, and their pixel-space controls do not provide a direct interface to JEPA latent predictors. We therefore use controllable video world models offline as sources of action-conditioned dynamics rewards for efficient latent predictors.

2.2 Generalization problems in Robot Policy

Generalization in embodied action-conditioned models. On the *policy* side, embodied generalization has scaled quickly[98]: task-conditioned imitation and robotics transformers [48, 16] grew into web-knowledge-grounded VLM [80, 99, 97, 101, 57] and VLA models [29, 15, 52, 97], and diffusion- and flow-based action heads now serve as standard expressive multimodal controllers [24, 11, 74, 104, 103]. Generalist policies further widen task and embodiment coverage via multi-task agents [14, 70, 102, 82], cross-embodiment transformers [27, 62] and video-based action pretraining [20, 32, 93, 99]. This scaling rides progressively larger embodied corpora [26, 86, 71, 51, 1] and complementary work on morphology, dexterity, and human-to-robot video transfer [73, 47, 6, 78, 43]. Yet action-conditioned *world models* have not seen comparable scaling: in JEPA-style latent world models, the encoder inherits priors from large-scale video while the conditional transition $(h, a) \to h'$ is still learned from a limited embodied corpus, leaving the predictor as the narrow component and motivating our study of generalization for the predictor itself.

Synthetic rollouts and reward-based supervision. A growing body of work expands embodied training through synthetic experience: from replay-based demonstration augmentation [67] and learned interactive simulators [92], through action-conditioned video world models [76, 45, 69] and synthetic-trajectory [49]. A complementary thread uses human-video pretraining for robot representations and rewards [68, 65], or derives rewards from generative-model likelihoods [46, 90, 63]. However, directly distilling generated futures is unreliable, since rollouts can be visually plausible yet physically inconsistent [108] and often fail closed-loop control [31, 105]; JEPAGeneralist therefore distills only action dynamics from them, leaving full latent prediction anchored to real video.

3 Methodology

We introduce JEPAGeneralist, a framework for training generalist action-conditioned JEPA-style latent world models with video-world-model motion rewards. The core idea is to use real videos to anchor source-conditioned future prediction, while using controllable video world models (VWMs) to provide same-action motion priors beyond the support of the embodied training data. Unlike direct generated-latent distillation, JEPAGeneralist does not treat generated futures as full future-latent targets. Instead, it factorizes future latent prediction into source-conditioned content continuation and action-induced dynamics, and uses generated rollouts only through their patch-level temporal deltas. We refer to this signal as a *VWM motion reward*: an offline reward-like supervision signal that encourages predicted temporal changes to match same-action generated rollouts, implemented as a deterministic temporal-delta loss rather than an online contrastive reward.

3.1 Action-Conditioned Latent World Modeling

Let $x_{\leq t}$ denote the observed video history and let $a_{t+1:t+T}$ denote a planned future action or control sequence. A frozen JEPA encoder E maps video frames into latent representations: $h_{\leq t} = E(x_{\leq t})$ and $h_{t+1:t+T} = E(x_{t+1:t+T})$. The action sequence is treated as an external control input, as in standard model-based prediction: given a candidate action sequence, the world model predicts the resulting future state. The goal is to learn an action-conditioned latent predictor $\hat{h}_{t+1:t+T} = P_\theta(h_{\leq t}, a_{t+1:t+T})$ that generalizes beyond the action, task, object, and embodiment categories observed during training.

We study this problem under a held-out embodied-category setting, instantiated in our experiments as held-out subtask generalization. Let $\mathcal{D}_{\text{tr}}$ and $\mathcal{D}_{\text{te}}$ denote training and testing distributions over latent-action-future triples: $(h_{\leq t}, a_{t+1:t+T}, h_{t+1:t+T}) \sim \mathcal{D}$. The predictor is trained on categories from $\mathcal{D}_{\text{tr}}$ and evaluated on disjoint categories from $\mathcal{D}_{\text{te}}$. This setting directly measures whether the latent dynamics model generalizes beyond the training action/task support.

Why the predictor is the bottleneck. A JEPA-style latent world model contains a visual encoder E and an action-conditioned predictor P_θ. Although E may benefit from broad video pretraining, P_θ is trained on a joint latent-action distribution whose support is determined by the embodied dataset, action representation, tasks, objects, and embodiments used for training. For example, an action-conditioned predictor may be post-trained on substantially narrower embodied data than the video data used to pretrain the visual encoder. Thus, generalization of the encoder over observations does not imply generalization of the conditional transition model over (h, a). Let $z = (h_{\leq t}, a_{t+1:t+T})$ and $y = h_{t+1:t+T}$. For a predictor $P \in \mathcal{H}$, define $R_{\mathcal{D}}(P) = \mathbb{E}_{(z,y)\sim\mathcal{D}}[\ell(P(z), y)]$. Under distribution shift, the test risk is controlled by the training risk plus a discrepancy term between $\mathcal{D}_{\text{tr}}(z)$ and $\mathcal{D}_{\text{te}}(z)$, as well as the best joint error of the hypothesis class. Therefore, even if P achieves low training error, its held-out risk can remain high when the training and testing latent-action distributions differ substantially. Moreover, if a held-out input lies outside the training support, its transition is not identifiable from the action-conditioned training data alone. This motivates using video world models as additional sources of action-conditioned dynamics priors.

Video world models as broader dynamics priors. Controllable video world models are trained on broader video distributions and expose flexible video-space control interfaces, such as skeletons, trajectories, masks, references, or VACE-style controls. We do not assume that these interfaces form a universal robot action space. Instead, we view them as a broader visual-control manifold that can express diverse action-like motion cues. However, these visual control signals still introduce distribution and modality gaps between latent world models and video world models, since pixel-space controls are often more natural for video synthesis than compact embodied action representations. By rendering planned action proxies into this interface, we can query a video world model for same-action rollouts across diverse visual environments. These rollouts provide additional action-conditioned dynamics priors beyond the narrow training support of the JEPA predictor.

3.2 Generated Futures Are Not Absolute Targets

A natural way to use a controllable video world model is to generate a future video $\tilde{x}_{t+1:t+T}$ under the same planned action, encode it using the frozen JEPA encoder, $\tilde{h}_{t+1:t+T} = E(\tilde{x}_{t+1:t+T})$, and directly distill the generated latent: $\mathcal{L}_{\text{gen-distill}} = D\Big(P_\theta(h_{\leq t}, a_{t+1:t+T}), \tilde{h}_{t+1:t+T}\Big)$. However, generated futures are not guaranteed to be valid continuations of the observed source context.

This issue is further amplified by the interface mismatch between video generation models and latent action-conditioned world models. General-purpose video generation models are often controlled through visual proxies, such as skeletons, trajectories, masks, or reference videos, whereas latent world-model predictors are conditioned on compact embodied action representations. As a result, the generated rollout may preserve the coarse intended motion while still being mismatched with the fine-grained robotics or embodied control signal used by the predictor. A no-reference or weakly referenced video world model may preserve the intended action while changing the environment, object identity, hand appearance, camera layout, or generation style. Thus, the generated latent may contain useful action dynamics, but it also contains source-mismatched content and generation artifacts. This mismatch can be further amplified by re-encoding generated videos with the frozen VJEPA encoder. Since the encoder is trained primarily on real videos, generation artifacts, temporal

inconsistency, or frame-rate mismatch may produce out-of-distribution latent targets and harm the latent world-model predictor.

To make this explicit, suppose a future latent can be conceptually decomposed into content/state continuation and action-induced dynamics: $h^{\mathrm{real}} = c^{\mathrm{src}} + d(a) + \epsilon$ and $\tilde{h} = c^{\mathrm{gen}} + d(a) + \eta$, where c^{src} denotes source-conditioned content continuation, c^{gen} denotes generated scene/content, $d(a)$ denotes action-induced dynamics, and ϵ, η denote residual noise or artifacts. Directly distilling $\tilde{h}$ forces the predictor to match not only $d(a)$ but also the source-mismatched content term c^{gen}. For a squared loss that combines real and generated targets, the optimum is biased toward

$$\frac{h^{\mathrm{real}} + \lambda \tilde{h}}{1+\lambda} = c^{\mathrm{src}} + d(a) + \frac{\lambda}{1+\lambda}(c^{\mathrm{gen}} - c^{\mathrm{src}}) + \mathrm{noise}. \tag{1}$$

Whenever $c^{\mathrm{gen}} \neq c^{\mathrm{src}}$, this introduces a source-mismatched content bias. This motivates our factorized predictor and VWM motion-reward design: generated videos should regularize temporal dynamics, rather than directly supervise the full future latent. We include this direct absolute generated-latent distillation as the GenAbs ablation in experiments.

3.3 JEPAGeneralist

Overview. JEPAGeneralist consists of three components: a content–dynamics factorized JEPA predictor, an offline VWM motion-reward bank, and a patch-level motion-reward distillation objective. Real future latents always supervise the full prediction. Generated video rollouts are not used as absolute targets or online contrastive rewards. Instead, their frozen-JEPA temporal deltas provide offline motion-reward supervision for the predicted future latent dynamics. This allows the predictor to benefit from video-world-model dynamics priors while avoiding direct supervision from source-mismatched generated content.

Content–Dynamics Factorized Predictor. Instead of predicting the future latent as a single entangled output, JEPAGeneralist factorizes prediction into a content branch and a dynamics branch. Given past latent states $h_{\leq t}$ and planned action sequence $a_{t+1:t+T}$, we compute $c_{t+1:t+T} = P_c(h_{\leq t})$ and $d_{t+1:t+T} = P_d(h_{\leq t}, a_{t+1:t+T})$, where P_c predicts source-conditioned content continuation and P_d predicts action-induced dynamics residual. The final future latent is produced by a learnable fusion module: $\hat{h}_{t+1:t+T} = F_\psi(c_{t+1:t+T}, d_{t+1:t+T}, h_{\leq t}, a_{t+1:t+T})$. In our main calibrated model, F_ψ is implemented as a content-anchored cross-attention fusion block. The content tokens serve as the source-conditioned anchor and attend to action-induced dynamics tokens, allowing the model to inject motion cues while preserving the calibrated JEPA latent manifold. We also evaluate a gated additive fusion variant in ablations, which injects dynamics more aggressively and improves action recoverability at the cost of latent calibration. The key principle is that content and action dynamics are represented separately before composition.

This factorization is central to our method. The full prediction $\hat{h}$ is supervised by real future latents, so the model remains anchored to the observed source context. The VWM motion reward is applied to the final predicted future latent dynamics through temporal deltas, preventing source-mismatched generated content from directly corrupting the absolute future latent target. The real supervised loss is $\mathcal{L}_{\mathrm{real}} = D(\hat{h}_{t+1:t+T}, h_{t+1:t+T})$, where D combines mean-squared error and cosine distance. This loss is always active.

Offline VWM Motion-Reward Bank. Directly querying a video generation model during predictor training would be computationally expensive, since video synthesis requires iterative denoising and high-dimensional pixel generation. It would also couple the latent predictor to the stochastic generation process. We therefore build an offline VWM motion-reward bank using a frozen controllable video world model and a frozen JEPA encoder. To obtain broader action-dynamics feedback and address the interface difference between video generation models and latent world models, we use a controllable video world model as a same-action environment randomizer. For each training sample, we render the planned action proxy into a visual control signal $u_{t+1:t+T} = r(a_{t+1:t+T})$, where $r(\cdot)$ may produce skeleton, trajectory, or other visual controls. A frozen controllable video world model then generates a same-action rollout $\tilde{x}_{t+1:t+T} \sim G_{\mathrm{VWM}}(\cdot \mid u_{t+1:t+T})$. This rollout may differ in scene, object appearance, or background, which provides environment randomization. Each generated future is encoded by the frozen JEPA encoder: $\tilde{h}_{t+1:t+T} = E(\tilde{x}_{t+1:t+T})$. For each training sample, we store the real past latent, real future latent, generated future latent, action sequence,

and metadata $(h_{\leq t}, h_{t+1:t+T}, \tilde{h}_{t+1:t+T}, a_{t+1:t+T})$. The generated latent is stored at patch level, i.e., $\tilde{h}_{t+1:t+T} \in \mathbb{R}^{T\times P\times D}$. The bank is built offline, so predictor training does not run video generation, video decoding, or generated-video JEPA encoding.

Patch-Level Motion-Reward Distillation. Although generated rollouts may contain useful same-action motion, their absolute latents may differ from the real future due to changes in scene content, object identity, camera layout, hand appearance, or generation artifacts. Therefore, we do not supervise the predictor with $\tilde{h}_{t+1:t+T}$ directly. Instead, we use only the temporal deltas of the generated JEPA latents as offline motion-reward targets. Given the predicted future latent $\hat{h}_{t+1:t+T} = P_\theta(h_{\leq t}, a_{t+1:t+T})$, we define patch-level temporal deltas as $\Delta\hat{h}_{t+i} = \hat{h}_{t+i+1} - \hat{h}_{t+i}$ and $\Delta\tilde{h}_{t+i} = \tilde{h}_{t+i+1} - \tilde{h}_{t+i}$, for $i = 1, \ldots, T-1$. The motion-reward distillation loss is $\mathcal{L}_{\mathrm{MR}} = D\Big(\Delta\hat{h}_{t+1:t+T-1}, \Delta\tilde{h}_{t+1:t+T-1}\Big)$, where D combines mean-squared error and cosine distance over patch-level latent tokens. This objective distills the motion evolution induced by the same-action generated rollout while avoiding absolute supervision from source-mismatched generated content. We denote this aligned generated temporal-delta signal as Aligned GenΔ.

Training Objective and Inference. The final objective is $\mathcal{L} = \mathcal{L}_{\mathrm{real}} + \lambda_{\mathrm{MR}}\mathcal{L}_{\mathrm{MR}}$. We use a real-only warmup before enabling the VWM motion reward, allowing the predictor to first learn a stable source-conditioned latent transition. In our best configuration, we set $\lambda_{\mathrm{MR}} = 0.1$ and enable $\mathcal{L}_{\mathrm{MR}}$ after a 10-epoch warmup. The best checkpoint is selected using held-out real future prediction metrics, not generated-video scores.

At inference time, JEPAGeneralist reduces to a lightweight action-conditioned latent predictor. Given an observed history and a planned action sequence, it predicts future JEPA latents directly. No video world model, generated-video decoder, or generated-video JEPA encoding is used during rollout.

Action-Recoverability Probe. To evaluate whether predicted latent futures preserve action-relevant dynamics, we train a schema-conditioned inverse dynamics probe on real JEPA latents. Given $h_{\leq t}$ and a future latent sequence, the probe predicts an action trajectory under a chosen action abstraction, such as hand center, fingertips, wrist-fingertips, three fingers, or full hand pose. This decoded trajectory is an evaluation signal for action recoverability; it should not be interpreted as the planned control input used by the predictor. The probe is frozen during evaluation and is used only as an analysis tool. It allows us to measure whether predicted future latents preserve recoverable action information, complementing latent prediction metrics.

4 Experiments

4.1 Settings

Datasets. We evaluate JEPAGeneralist on three complementary benchmarks: **EgoDex** [43], **BAIR Robot Pushing** [33], and **Physion** [10]. EgoDex is our primary held-out cross-task generalization benchmark, where models are evaluated on three unseen subtasks: `assemble square table`, `fold`, and `insert round table`. BAIR evaluates future-frame latent prediction in the V-JEPA latent space. Physion evaluates whether predicted future latents preserve physically relevant contact information. Full dataset protocols and split details are provided in Appendix E.

Baselines. On EgoDex, we compare against frozen-backbone latent predictor baselines: **V-JEPA2**, **V-JEPA2 AC**, **DINOv2**, and **DINO-WM**. On BAIR, we compare against **V-JEPA + MLP** and **V-JEPA Predictor**. On Physion, we compare against **DINO-WM**, **V-JEPA observed-only**, and **V-JEPA AC Predictor**. For ablations, we evaluate **Basic**, **GenAbs**, **Factorized**, **Cross-Attn**, **GatedAdd**, **Aligned GenΔ**, **Shuffled GenΔ**, and **Real-Δ Aux**. Unless otherwise specified, JEPAGeneralist denotes our main calibrated variant with content–dynamics factorization, Cross-Attn fusion, and Aligned GenΔ motion-reward distillation. Detailed baseline definitions are provided in Appendix E.2, and variant definitions in Appendix C.

Metrics. We report latent prediction metrics, action sensitivity metrics, action recoverability metrics, and physical contact readout metrics. Latent prediction is evaluated by **MSE**, **cosine similarity**, **R@1**, **R@5**, and temporal latent-delta error **ΔErr**. Action sensitivity is measured by **Shuffle Sensitivity** and **Zero Sensitivity**. Action recoverability is measured by **ADE**, **FDE**, **PCK@0.05**, and velocity consistency using the schema-conditioned IDM probe. Physion contact readout is evaluated by

Table 1: External latent-space baselines on EgoDex. DINO-based models operate in a different latent space from V-JEPA-style models, so raw MSE and cosine similarity are not directly comparable; we mark these entries as N/A and emphasize IDM-based action recoverability metrics for cross-latent-space comparison.

Method	Latent Space	Predictor	MSE↓	Cos↑	ADE↓	FDE↓	PCK↑
V-JEPA2 AC	V-JEPA2	standard	5.93	0.65	0.10	0.10	0.33
V-JEPA2	V-JEPA2	MLP	6.42	0.62	0.10	**0.09**	0.29
DINO-WM	DINOv2	standard	N/A	N/A	0.11	0.11	0.24
DINOv2	DINOv2	MLP	N/A	N/A	0.12	0.11	0.24
JEPAGeneralist	V-JEPA2	standard	**5.92**	**0.66**	**0.09**	**0.09**	**0.40**

Table 2: Physion contact readout results. We report Contact Accuracy, Balanced Accuracy, and ROC-AUC under the contact/OCP readout protocol.

Method	Contact Acc↑	Balanced Acc↑	ROC-AUC↑
DINO-WM	0.6483	0.6483	0.7049
V-JEPA observed-only	0.7383	0.7383	0.8201
V-JEPA AC Predictor	0.7408	0.7408	0.8213
JEPAGeneralist	**0.7817**	**0.7817**	**0.8622**

Table 3: Per-subtask action recoverability on three unseen EgoDex subtasks: `assemble square table`, `fold`, and `insert round table`. Each subtask reports ADE↓, FDE↓, and PCK@0.05↑.

Method	Assemble Square Table			Fold			Insert Round Table		
	ADE ↓	FDE ↓	PCK ↑	ADE ↓	FDE ↓	PCK ↑	ADE ↓	FDE ↓	PCK ↑
DINOv2 + AC MLP	0.0572	0.0558	0.4501	0.2084	0.1840	0.0256	0.0689	0.0659	0.3370
V-JEPA2 AC	0.0516	0.0516	0.5707	0.1895	0.1833	**0.0272**	0.0610	0.0594	0.4081
DINO-WM	0.0570	0.0559	0.4493	0.1987	0.1786	0.0192	0.0658	0.0634	0.3512
JEPAGeneralist	**0.0453**	**0.0446**	**0.6800**	**0.1842**	**0.1740**	**0.0272**	**0.0601**	**0.0593**	**0.4428**

Table 4: BAIR robot pushing results in the V-JEPA latent space. Shuffle and zero sensitivities measure whether replacing or removing the action input degrades future latent prediction.

Method	MSE↓	Cos↑	R@1↑	R@5↑	Shuffle Sens↑	Zero Sens↑
V-JEPA + MLP	5.3124	0.6653	0.2188	0.9049	0.0274	0.0441
V-JEPA Predictor	4.6027	0.7178	0.2474	0.9974	0.0000	0.0000
JEPAGeneralist	4.4828	0.7271	0.3464	**0.9987**	0.1910	0.1571
JEPAGeneralist w/Aux	**4.4782**	**0.7272**	**0.3581**	**0.9987**	**0.2078**	**0.1655**

Contact Accuracy, **Balanced Accuracy**, and **ROC-AUC**. Metric definitions and implementation details are provided in Appendix E.3.

4.2 Experimental Results

EgoDex Cross-Task Generalization. EgoDex is our primary held-out generalization benchmark. Models are trained on seen embodied action/task categories and evaluated on three disjoint held-out subtasks. Table 1 compares frozen-backbone latent predictors, and Table 3 further breaks down action recoverability by held-out subtask.

Within the V-JEPA2 latent space, the standard action-conditioned Transformer improves over a shallow MLP predictor, confirming that a strong frozen encoder alone is insufficient. Across latent backbones, V-JEPA2-based predictors preserve more action-recoverable dynamics than DINOv2-based predictors. JEPAGeneralist achieves the best overall action recoverability, improving PCK from 0.33 to 0.40 over the V-JEPA2 AC baseline, showing that VWM motion rewards improve the action-relevant structure of predicted future latents. JEPAGeneralist improves ADE, FDE, and PCK on all reported held-out subtasks, indicating stronger recoverable action dynamics across unseen tasks. The `fold` subtask remains the most challenging, with substantially larger displacement errors than assembly or insertion subtasks, motivating per-subtask reporting in addition to aggregate metrics.

Table 5: Component ablation on EgoDex with latent prediction, action sensitivity, and IDM-based action recoverability metrics. ΔErr denotes temporal latent-delta error. Aligned GenΔ is our offline VWM motion reward. The two selected variants are placed at the bottom: JEPAGeneralist (Cross-Attn) for calibrated latent prediction and JEPAGeneralist-GatedAdd for action recoverability. Bold indicates the best result.

Components	Aux. λ	MSE↓	Cos↑	R@1↑	ΔErr↓	Shuffle↑	Zero↑	ADE↓	FDE↓	PCK@.05↑
Basic	–	5.9312	0.6540	0.8178	11.214	0.1011	0.1058	0.0973	0.0983	0.3229
Basic + GenAbs	0.1	5.9240	0.6547	0.8323	11.216	0.1059	0.1161	0.0963	0.0975	0.3232
Basic + Factorized + Cross-Attn	–	5.9424	0.6534	0.7516	11.207	0.0550	0.0814	0.0973	0.0948	0.3922
Basic + Factorized + Cross-Attn + Aligned GenΔ	0.2	5.9287	0.6546	0.7536	11.177	0.0814	0.1022	0.0968	0.0930	0.3689
Basic + Factorized + Cross-Attn + Shuffled GenΔ	0.1	5.9325	0.6546	0.7702	11.194	0.0785	0.1111	0.0972	0.0941	0.3636
Basic + Factorized + Cross-Attn + Real-Δ Aux	0.1	5.9276	0.6541	0.7536	11.190	0.0439	0.0650	0.1004	0.0963	0.3626
JEPAGeneralist (Cross-Attn)	**0.1**	**5.9177**	**0.6552**	0.7743	**11.172**	0.0846	0.0825	0.0938	0.0948	0.3526
JEPAGeneralist-GatedAdd	**0.1**	6.2485	0.6310	**0.8571**	11.768	**0.1815**	**0.2117**	**0.0931**	**0.0909**	**0.4222**

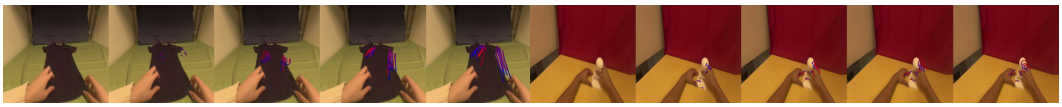


Figure 2: **Qualitative visualization of predicted future hand trajectories.** We decode future hand trajectories from predicted latents using the schema-conditioned IDM probe and overlay the resulting 3D joints on a reference frame. The color gradient indicates temporal progression, from red at the trajectory start to blue at the final endpoint. Compared with DINO-WM, V-JEPA, and V-JEPA AC, JEPAGeneralist produces smoother trajectories and blue endpoints that better align with the expected hand-object interaction, indicating stronger held-out action generalization.

Component Ablation. Table 5 ablates the core design choices of JEPAGeneralist. Aligned GenΔ denotes our offline VWM motion reward, and detailed variant definitions are provided in Appendix C. GenAbs improves some latent metrics but does not improve temporal-delta error, supporting our claim that absolute generated futures are not ideal targets. Cross-Attn with Aligned GenΔ achieves the best MSE, cosine similarity, and ΔErr, and is therefore used as the main calibrated variant. JEPAGeneralist-GatedAdd achieves the best action recoverability and sensitivity metrics, revealing a calibration–recoverability trade-off. Shuffled GenΔ and Real-Δ Aux underperform aligned VWM motion rewards, indicating that the gains are not explained by generic temporal smoothing alone.

BAIR Robot Pushing. On BAIR, JEPAGeneralist reduces MSE from 4.6027 to 4.4828 over the V-JEPA Predictor baseline and improves R@1 from 0.2474 to 0.3464. Adding a lightweight content-smoothing auxiliary term further improves MSE to 4.4782 and R@1 to 0.3581. The gains in shuffle and zero sensitivity show that JEPAGeneralist uses the action input more effectively than the real-only baseline.

Physion Contact Readout. JEPAGeneralist achieves the best result across all Physion contact readout metrics. It improves Contact Accuracy from 0.7408 to 0.7817 over the V-JEPA AC Predictor and improves ROC-AUC from 0.8213 to 0.8622, suggesting that its predicted future latents better preserve physically relevant contact information.

Qualitative Action-Trajectory Visualization. Figure 2 visualizes future hand trajectories decoded from predicted latents using the schema-conditioned IDM probe and overlaid on a reference frame. The color gradient denotes temporal progression, from red at the trajectory start to blue at the final endpoint. Compared with DINO-WM, V-JEPA, and V-JEPA AC, JEPAGeneralist produces smoother and more temporally coherent trajectories, and its blue endpoints align more closely with the expected hand-object interaction. This qualitative trend matches the quantitative improvements in ADE, FDE, and PCK, suggesting that VWM motion-reward supervision improves held-out action generalization rather than merely reducing latent prediction error.

4.3 Broader Task Generalization on EgoDex

We extend evaluation to the complete official EgoDex test split [43]: 3,243 episodes across 111 tasks, including 2,525 episodes from 102 tasks absent from our predictor's training set. We evaluate frozen checkpoints without retraining, test-time adaptation, or test-based model selection. Metrics are

averaged within each task and then equally across all 111 tasks. Table 6 shows that JEPAGeneralist achieves the lowest macro ADE and FDE while essentially matching the strongest PCK, extending evaluation beyond the original three held-out subtasks. These results describe the full test set, including both seen and unseen task labels.

Table 6: Action recoverability on all 111 official EgoDex test tasks, including 102 training-unseen tasks. Results are equal-task macro averages; PCK uses a threshold of 0.05.

Method	ADE↓	FDE↓	PCK↑
V-JEPA2 AC [4]	0.090111	0.090360	0.295200
DINO-WM [106]	0.112140	0.115215	0.153427
DINOv2 [72]	0.116268	0.119611	0.146093
V-JEPA2 [4]	0.091537	0.092357	0.270015
JEPAGeneralist	**0.089947**	**0.089639**	**0.295222**

4.4 Effect of the Auxiliary Supervision Source

To isolate the contribution of generated motion supervision from VACE [50], we conduct a separate source-control audit with the same factorized architecture and a fixed 483-trajectory evaluation protocol. We compare no auxiliary, real-future temporal deltas, source-shuffled generated deltas, and correctly aligned generated deltas over three training seeds. Shuffling preserves the generated bank but breaks source–control pairing. As shown in Table 7, aligned GenΔ improves action sensitivity and recovery over Real-Δ and source-shuffled GenΔ. The Shuffle, ADE, and FDE improvements over both controls hold in all three seeds. Real-Δ has a larger measured output-gradient ratio (0.2329 versus 0.1848), arguing against auxiliary strength alone explaining the gains. However, aligned GenΔ increases MSE and temporal Δ error by 0.86% and 1.11% relative to Real-Δ: the benefit is improved action-relevant structure, rather than uniform metric dominance.

Table 7: Auxiliary-source audit on the fixed 483-trajectory evaluation set. Mean $\pm$ sample standard deviation over seeds 42/43/44; this is a separate protocol from the 111-task evaluation.

Auxiliary source	Shuffle↑	Zero↑	ADE↓	FDE↓
None	.1045±.0290	.0853±.0241	.0950±.0005	.0923±.0014
Real-Δ	.1114±.0173	.0888±.0153	.0953±.0011	.0920±.0014
Source-shuffled GenΔ	.1218±.0144	.0952±.0100	.0956±.0002	.0930±.0003
Aligned GenΔ	**.1558**±.0317	**.1133**±.0357	**.0936**±.0007	**.0905**±.0008

5 Conclusion

We presented JEPAGeneralist, a generalist action-conditioned JEPA framework that transfers broad motion priors from frozen video world models into efficient latent predictors. Rather than using generated videos as absolute targets, JEPAGeneralist anchors prediction with real future latents and distills only patch-level temporal deltas from same-action generated rollouts as offline VWM motion rewards. This design preserves the efficiency of JEPA latent rollout while leveraging the scale and diversity of controllable video world models. Across EgoDex, BAIR, and Physion, JEPAGeneralist improves latent prediction, action sensitivity, action recoverability, and physical contact readout under held-out generalization settings. Together, these results suggest that frozen video world models can serve as scalable offline teachers for more generalist latent predictors beyond narrow embodied training distributions. Looking forward, we hope this direction helps pave the way toward addressing the long-standing bottleneck in scaling up action-conditioned latent world models.

6 Acknowledgements

This work was conducted at Northeastern University and was supported by Northeastern University's Physical AI Research Initiative (PAIR).

References

[1] AgiBot-World Contributors, Bu, Q., Cai, J., Chen, L., Cui, X., Ding, Y., Feng, S., Gao, S., He, X., Huang, X., et al.: AgiBot World Colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. arXiv preprint arXiv:2503.06669 (2025)

[2] Assran, M., Duval, Q., Misra, I., Bojanowski, P., Vincent, P., Rabbat, M., LeCun, Y., Ballas, N.: Self-supervised learning from images with a joint-embedding predictive architecture. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 15619–15629. IEEE (2023)

[3] Assran, M., Duval, Q., Misra, I., Bojanowski, P., Vincent, P., Rabbat, M., LeCun, Y., Ballas, N.: Self-supervised learning from images with a joint-embedding predictive architecture. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 15619–15629 (2023)

[4] Assran, M., Bardes, A., Fan, D., Garrido, Q., Howes, R., Muckley, M., Rizvi, A., Roberts, C., Sinha, K., Zholus, A., et al.: V-jepa 2: Self-supervised video models enable understanding, prediction and planning. arXiv preprint arXiv:2506.09985 (2025)

[5] Assran, M., Bardes, A., Fan, D., Garrido, Q., Howes, R., Muckley, M., Rizvi, A., Roberts, C., Sinha, K., Zholus, A., et al.: V-jepa 2: Self-supervised video models enable understanding, prediction and planning. arXiv preprint arXiv:2506.09985 (2025)

[6] Bahl, S., Gupta, A., Pathak, D.: Human-to-robot imitation in the wild. In: Robotics: Science and Systems (2022)

[7] Baldassarre, F., Szafraniec, M., Terver, B., Khalidov, V., Massa, F., LeCun, Y., Labatut, P., Seitzer, M., Bojanowski, P.: Back to the features: Dino as a foundation for video world models. arXiv preprint arXiv:2507.19468 (2025)

[8] Ball, P.J., Bauer, J., Belletti, F., Brownfield, B., Ephrat, A., Fruchter, S., Gupta, A., Holsheimer, K., Holynski, A., Hron, J., Kaplanis, C., Limont, M., McGill, M., Oliveira, Y., Parker-Holder, J., Perbet, F., Scully, G., Shar, J., Spencer, S., Tov, O., Villegas, R., Wang, E., Yung, J., Baetu, C., Berbel, J., Bridson, D., Bruce, J., Buttimore, G., Chakera, S., Chandra, B., Collins, P., Cullum, A., Damoc, B., Dasagi, V., Gazeau, M., Gbadamosi, C., Han, W., Hirst, E., Kachra, A., Kerley, L., Kjems, K., Knoepfel, E., Koriakin, V., Lo, J., Lu, C., Mehring, Z., Moufarek, A., Nandwani, H., Oliveira, V., Pardo, F., Park, J., Pierson, A., Poole, B., Ran, H., Salimans, T., Sanchez, M., Saprykin, I., Shen, A., Sidhwani, S., Smith, D., Stanton, J., Tomlinson, H., Vijaykumar, D., Wang, L., Wingfield, P., Wong, N., Xu, K., Yew, C., Young, N., Zubov, V., Eck, D., Erhan, D., Kavukcuoglu, K., Hassabis, D., Gharamani, Z., Hadsell, R., van den Oord, A., Mosseri, I., Bolton, A., Singh, S., Rocktäschel, T.: Genie 3: A new frontier for world models (2025), `https://deepmind.google/blog/genie-3-a-new-frontier-for-world-models/`

[9] Bardes, A., Garrido, Q., Ponce, J., Chen, X., Rabbat, M., LeCun, Y., Assran, M., Ballas, N.: Revisiting feature prediction for learning visual representations from video. Transactions on Machine Learning Research (2024)

[10] Bear, D., Wang, E., Mrowca, D., Binder, F.J., Tung, H.Y., Pramod, R., Holdaway, C., Tao, S., Smith, K.A., Sun, F.Y., et al.: Physion: Evaluating physical prediction from vision in humans and machines. In: Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1) (2021)

[11] Black, K., Brown, N., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., Groom, L., Hausman, K., Ichter, B., et al.: π_0: A vision-language-action flow model for general robot control. arXiv preprint arXiv:2410.24164 (2024)

[12] Blattmann, A., Dockhorn, T., Kulal, S., Mendelevitch, D., Kilian, M., Lorenz, D., Levi, Y., English, Z., Voleti, V., Letts, A., et al.: Stable video diffusion: Scaling latent video diffusion models to large datasets. arXiv preprint arXiv:2311.15127 (2023)

[13] Blattmann, A., Rombach, R., Ling, H., Dockhorn, T., Kim, S.W., Fidler, S., Kreis, K.: Align your latents: High-resolution video synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 22563–22575 (2023)

[14] Bousmalis, K., Vezzani, G., Rao, D., Devin, C., Lee, A.X., Bauza, M., Davchev, T., Zhou, Y., Gupta, A., Raju, A., Laurens, A., Fantacci, C., Dalibard, V., Zambelli, M., Martins, M., Pevceviciute, R., Blokzijl, M., Denil, M., Batchelor, N., Lampe, T., Parisotto, E., Zolna, K., Reed, S., Colmenarejo, S.G., Scholz, J., Abdolmaleki, A., Groth, O., Regli, J.B., Sushkov, O., Rothorl, T., Chen, J.E., Aytar, Y., Barker, D., Ortiz, J., Riedmiller, M., Springenberg, J.T., Hadsell, R., Nori, F., Heess, N.: Robocat: A self-improving generalist agent for robotic manipulation. Transactions on Machine Learning Research (2023)

[15] Brohan, A., Brown, N., Carbajal, J., Chebotar, Y., Chen, X., Choromanski, K., Ding, T., Driess, D., Dubey, A., Finn, C., et al.: Rt-2: Vision-language-action models transfer web knowledge to robotic control. arxiv. arXiv preprint arXiv:2307.15818 (2023)

[16] Brohan, A., Brown, N., Carbajal, J., Chebotar, Y., Dabis, J., Finn, C., Gopalakrishnan, K., Hausman, K., Herzog, A., Hsu, J., et al.: Rt-1: Robotics transformer for real-world control at scale. arXiv preprint arXiv:2212.06817 (2022)

[17] Brooks, T., Peebles, B., Holmes, C., DePue, W., Guo, Y., Jing, L., Schnurr, D., Taylor, J., Luhman, T., Luhman, E., et al.: Video generation models as world simulators. OpenAI Blog **1**(8), 1 (2024)

[18] Bruce, J., Dennis, M.D., Edwards, A., Parker-Holder, J., Shi, Y., Hughes, E., Lai, M., Mavalankar, A., Steigerwald, R., Apps, C., et al.: Genie: Generative interactive environments. In: Forty-first International Conference on Machine Learning (2024)

[19] Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuscenes: A multimodal dataset for autonomous driving. In: 2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR). pp. 11618–11628. IEEE (2020)

[20] Cheang, C.L., Chen, G., Jing, Y., Kong, T., Li, H., Li, Y., Liu, Y., Wu, H., Xu, J., Yang, Y., Zhang, H., Zhu, M.: GR-2: A generative video-language-action model with web-scale knowledge for robot manipulation. arXiv preprint arXiv:2410.06158 (2024)

[21] Chen, B., Xu, Z., Kirmani, S., Ichter, B., Sadigh, D., Guibas, L., Xia, F.: Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 14455–14465 (2024)

[22] Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on pattern analysis and machine intelligence **40**(4), 834–848 (2017)

[23] Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International conference on machine learning. pp. 1597–1607. PmLR (2020)

[24] Chi, C., Xu, Z., Feng, S., Cousineau, E., Du, Y., Burchfiel, B., Tedrake, R., Song, S.: Diffusion policy: Visuomotor policy learning via action diffusion. In: Robotics: Science and Systems (2023)

[25] Chi, C., Xu, Z., Feng, S., Cousineau, E., Du, Y., Burchfiel, B., Tedrake, R., Song, S.: Diffusion policy: Visuomotor policy learning via action diffusion. The International Journal of Robotics Research **44**(10-11), 1684–1704 (2025)

[26] Dasari, S., Ebert, F., Tian, S., Nair, S., Bucher, B., Schmeckpeper, K., Singh, S., Levine, S., Finn, C.: Robonet: Large-scale multi-robot learning. arXiv preprint arXiv:1910.11215 (2019)

[27] Doshi, R., Walke, H., Mees, O., Dasari, S., Levine, S.: Scaling cross-embodied learning: One policy for manipulation, navigation, locomotion and aviation. In: Conference on Robot Learning (2024)

[28] Dosovitskiy, A.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)

[29] Driess, D., Xia, F., Sajjadi, M.S.M., Lynch, C., Chowdhery, A., Ichter, B., Wahid, A., Tompson, J., Vuong, Q., Yu, T., et al.: Palm-e: An embodied multimodal language model. arXiv preprint arXiv:2303.03378 (2023)

[30] Drozdov, K., Shwartz-Ziv, R., LeCun, Y.: Video representation learning with joint-embedding predictive architectures. arXiv preprint arXiv:2412.10925 (2024)

[31] Fan, C.K., Chi, X., Ju, X., Li, H., Bao, Y., Wang, Y.K., Chen, L., Jiang, Z., Ge, K., Li, Y., Mi, W., Wuwu, Q., Jia, P., Luo, Y., Zhang, K., Qin, Z., Dai, Y., Han, S., Guo, Y., Zhang, S., Tang, J.: Wow, wo, val! a comprehensive embodied world model evaluation Turing test. arXiv preprint arXiv:2601.04137 (2026)

[32] Feng, Y., Li, Y., Zhang, W., Zheng, S., Luo, H., Yue, Z., Lu, Z.: Videoorion: Tokenizing object dynamics in videos. In: 2025 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 20401–20412. IEEE (2025)

[33] Finn, C., Goodfellow, I., Levine, S.: Unsupervised learning for physical interaction through video prediction. Advances in neural information processing systems **29** (2016)

[34] Fu, X., Liu, S., Xu, Y., Lu, P., Hu, G., Yang, T., Anantasagar, T., Shen, C., Mao, Y., Liu, Y., et al.: Learning human-perceived fakeness in ai-generated videos via multimodal llms. arXiv preprint arXiv:2509.22646 (2025)

[35] Furukawa, Y., Ponce, J.: Accurate, dense, and robust multiview stereopsis. IEEE transactions on pattern analysis and machine intelligence **32**(8), 1362–1376 (2009)

[36] Geng, D., Herrmann, C., Hur, J., Cole, F., Zhang, S., Pfaff, T., Lopez-Guevara, T., Doersch, C., Aytar, Y., Rubinstein, M., et al.: Motion prompting: Controlling video generation with motion trajectories. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2025)

[37] Gillman, N., Herrmann, C., Freeman, M., Aggarwal, D., Luo, E., Sun, D., Sun, C.: Force prompting: Video generation models can learn and generalize physics-based control signals. arXiv preprint arXiv:2505.19386 (2025)

[38] Girshick, R.: Fast r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 1440–1448 (2015)

[39] Google DeepMind: Veo. `https://deepmind.google/models/veo/` (2026), accessed: 2026-05-05

[40] HaCohen, Y., Chiprut, N., Brazowski, B., Shalem, D., Moshe, D., Richardson, E., Levin, E., Shiran, G., Zabari, N., Gordon, O., et al.: Ltx-video: Realtime video latent diffusion. arXiv preprint arXiv:2501.00103 (2024)

[41] He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: 2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR). pp. 9726–9735. IEEE (2020)

[42] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)

[43] Hoque, R., Huang, P., Yoon, D.J., Sivapurapu, M., Zhang, J.: Egodex: Learning dexterous manipulation from large-scale egocentric video. arXiv preprint arXiv:2505.11709 (2025)

[44] Hu, Y., Yang, J., Chen, L., Li, K., Sima, C., Zhu, X., Chai, S., Du, S., Lin, T., Wang, W., et al.: Planning-oriented autonomous driving. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 17853–17862. IEEE (2023)

[45] Huang, S., Wu, J., Zhou, Q., Miao, S., Long, M.: Vid2World: Crafting video diffusion models to interactive world models. In: International Conference on Learning Representations (2026)

[46] Huang, T., Jiang, G., Ze, Y., Xu, H.: Diffusion reward: Learning rewards via conditional video diffusion. In: European Conference on Computer Vision (2024)

[47] Huang, W., Mordatch, I., Abbeel, P., Pathak, D.: Generalization in dexterous manipulation via geometry-aware multi-task learning. arXiv preprint arXiv:2111.03062 (2021)

[48] Jang, E., Irpan, A., Khansari, M., Kappler, D., Ebert, F., Lynch, C., Levine, S., Finn, C.: Bc-z: Zero-shot task generalization with robotic imitation learning. In: Conference on Robot Learning. pp. 991–1002. PMLR (2022)

[49] Jang, J., Ye, S., Lin, Z., Xiang, J., Bjorck, J., Fang, Y., Hu, F., Huang, S., Kundalia, K., Lin, Y.C., Magne, L., Mandlekar, A., Narayan, A., Tan, Y.L., Wang, G., Wang, J., Wang, Q., Xu, Y., Zeng, X., Zheng, K., Zheng, R., Liu, M.Y., Zettlemoyer, L., Fox, D., Kautz, J., Reed, S., Zhu, Y., Fan, L.: DreamGen: Unlocking generalization in robot learning through video world models. arXiv preprint arXiv:2505.12705 (2025)

[50] Jiang, Z., Han, Z., Mao, C., Zhang, J., Pan, Y., Liu, Y.: Vace: All-in-one video creation and editing. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 17191–17202 (2025)

[51] Khazatsky, A., Pertsch, K., Nair, S., Balakrishna, A., Dasari, S., Karamcheti, S., Nasiriany, S., Srirama, M.K., Chen, L.Y., Ellis, K., et al.: Droid: A large-scale in-the-wild robot manipulation dataset. arXiv preprint arXiv:2403.12945 (2024)

[52] Kim, M.J., Pertsch, K., Karamcheti, S., Xiao, T., Balakrishna, A., Nair, S., Rafailov, R., Foster, E.P., Sanketi, P.R., Vuong, Q., et al.: Openvla: An open-source vision-language-action model. In: Conference on Robot Learning. pp. 2679–2713. PMLR (2025)

[53] Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A.A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., et al.: Overcoming catastrophic forgetting in neural networks. Proceedings of the national academy of sciences **114**(13), 3521–3526 (2017)

[54] Kong, W., Tian, Q., Zhang, Z., Min, R., Dai, Z., Zhou, J., Xiong, J., Li, X., Wu, B., Zhang, J., et al.: Hunyuanvideo: A systematic framework for large video generative models. arXiv preprint arXiv:2412.03603 (2024)

[55] LeCun, Y., et al.: A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. Open Review **62**(1), 1–62 (2022)

[56] Levine, S., Pastor, P., Krizhevsky, A., Ibarz, J., Quillen, D.: Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. The International journal of robotics research **37**(4-5), 421–436 (2018)

[57] Li, B., Shen, Y., Liu, Y., Xu, Y., Liu, J., Li, X., Li, Z., Zhu, J., Zhong, Y., Lan, F., et al.: Toward cognitive supersensing in multimodal large language model. arXiv preprint arXiv:2602.01541 (2026)

[58] Li, Y., Cai, W., Gao, Y., Li, C., Hu, X.: More than encoder: Introducing transformer decoder to upsample. In: 2022 IEEE international conference on bioinformatics and biomedicine (BIBM). pp. 1597–1602. IEEE (2022)

[59] Li, Y., Gao, Q., Zhao, T., Wang, B., Sun, H., Lyu, H., Hawkins, R.D., Vasconcelos, N., Golan, T., Luo, D., et al.: Core knowledge deficits in multi-modal language models. arXiv preprint arXiv:2410.10855 (2024)

[60] Li, Y., Wang, B., Vasconcelos, N.: Learning to think and explore in 3d world. arXiv preprint arXiv:2610.21783 (2026)

[61] Li, Y., Wang, X., Yang, L., Feng, L., Zhang, W., Gao, Y.: Diverse cotraining makes strong semi-supervised segmentor. arXiv preprint arXiv:2308.09281 (2023)

[62] Liu, M., Pathak, D., Agarwal, A.: Locoformer: Generalist locomotion via long-context adaptation. arXiv preprint arXiv:2509.23745 (2025)

[63] Liu, Y., Zhu, J., Mo, Y., Li, G., Cao, X., Jin, J., Shen, Y., Li, Z., Yu, T., Yuan, W., et al.: Palm: Progress-aware policy learning via affordance reasoning for long-horizon robotic manipulation. arXiv preprint arXiv:2601.07060 (2026)

[64] Ma, X., Yong, S., Zheng, Z., Li, Q., Liang, Y., Zhu, S.C., Huang, S.: Sqa3d: Situated question answering in 3d scenes. arXiv preprint arXiv:2210.07474 (2022)

[65] Ma, Y.J., Sodhani, S., Jayaraman, D., Bastani, O., Kumar, V., Zhang, A.: VIP: Towards universal visual reward and representation via value-implicit pre-training. In: International Conference on Learning Representations (2023)

[66] Ma, Y., Song, Z., Zhuang, Y., Hao, J., King, I.: A survey on vision–language–action models for embodied ai. IEEE Transactions on Neural Networks and Learning Systems (2026)

[67] Mandlekar, A., Nasiriany, S., Wen, B., Akinola, I., Narang, Y., Fan, L., Zhu, Y., Fox, D.: MimicGen: A data generation system for scalable robot learning using human demonstrations. In: Conference on Robot Learning (2023)

[68] Nair, S., Rajeswaran, A., Kumar, V., Finn, C., Gupta, A.: R3M: A universal visual representation for robot manipulation. In: Proceedings of The 6th Conference on Robot Learning. Proceedings of Machine Learning Research, vol. 205, pp. 892–909. PMLR (2023)

[69] NVIDIA, Agarwal, N., Ali, A., Bala, M., Balaji, Y., Barker, E., Cai, T., Chattopadhyay, P., Chen, Y., et al.: Cosmos world foundation model platform for physical AI. arXiv preprint arXiv:2501.03575 (2025)

[70] Octo Model Team, Ghosh, D., Walke, H., Pertsch, K., Black, K., Mees, O., Dasari, S., Hejna, J., Kreiman, T., Xu, C., et al.: Octo: An open-source generalist robot policy. arXiv preprint arXiv:2405.12213 (2024)

[71] Open X-Embodiment Collaboration, O'Neill, A., Rehman, A., Gupta, A., Maddukuri, A., Gupta, A., Padalkar, A., Lee, A., Pooley, A., Gupta, A., et al.: Open x-embodiment: Robotic learning datasets and rt-x models. arXiv preprint arXiv:2310.08864 (2023)

[72] Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. Transactions on Machine Learning Research Journal (2024)

[73] Pathak, D., Lu, C., Darrell, T., Isola, P., Efros, A.A.: Learning to control self-assembling morphologies: A study of generalization via modularity. In: Advances in Neural Information Processing Systems (2019)

[74] Physical Intelligence, Black, K., Brown, N., Darpinian, J., Dhabalia, K., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., et al.: $\pi_{0.5}$: A vision-language-action model with open-world generalization. arXiv preprint arXiv:2504.16054 (2025)

[75] Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 779–788 (2016)

[76] Rigter, M., Gupta, T., Hilmkil, A., Ma, C.: AVID: Adapting video diffusion models to world models. Reinforcement Learning Journal **6**, 737–764 (2025)

[77] Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015)

[78] Shaw, K., Bahl, S., Pathak, D.: Videodex: Learning dexterity from internet videos. In: Conference on Robot Learning (2023)

[79] Shen, Y., Liu, J., Li, X., Liu, Y., Li, B., Yang, H., Jia, W., Li, Y., Yu, T., Rehg, J.M., et al.: Egoforge: Goal-directed egocentric world simulator. arXiv preprint arXiv:2603.20169 (2026)

[80] Shen, Y., Liu, Y., Zhu, J., Cao, X., Zhang, X., He, Y., Ye, W., Rehg, J., Lourentzou, I.: Fine-grained preference optimization improves spatial reasoning in vlms. Advances in Neural Information Processing Systems **38**, 17929–17960 (2026)

[81] Shen, Y., Xu, J., Li, B., Zhang, Y., Yu, T., Li, B., Yang, H., Wang, R., Cao, X.: Chronovision: Temporal reasoning via latent state reconstruction. arXiv preprint arXiv:2608.05631 (2026)

[82] Shen, Y., Zhang, J., Xu, J., Kim, J., Lourentzou, I., Cao, X., Huang, M.: Evaluating cognitive age alignment in interactive ai agents. arXiv preprint arXiv:2605.17894 (2026)

[83] Shin, J., Li, Z., Zhang, R., Zhu, J.Y., Park, J., Schechtman, E., Huang, X.: Motionstream: Real-time video generation with interactive motion controls. arXiv preprint arXiv:2511.01266 (2025)

[84] Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., Abbeel, P.: Domain randomization for transferring deep neural networks from simulation to the real world. In: 2017 IEEE/RSJ international conference on intelligent robots and systems (IROS). pp. 23–30. IEEE (2017)

[85] Valevski, D., Leviathan, Y., Arar, M., Fruchter, S.: Diffusion models are real-time game engines. arXiv preprint arXiv:2408.14837 (2024)

[86] Walke, H., Black, K., Lee, A., Kim, M.J., Du, M., Zheng, C., Zhao, T., Hansen-Estruch, P., Vuong, Q., He, A., et al.: Bridgedata v2: A dataset for robot learning at scale. In: Conference on Robot Learning. pp. 1723–1736. PMLR (2023)

[87] Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.W., Chen, D., Yu, F., Zhao, H., Yang, J., et al.: Wan: Open and advanced large-scale video generative models. arXiv preprint arXiv:2503.20314 (2025)

[88] Wang, C., Chen, C., Huang, Y., Dou, Z., Liu, Y., Gu, J., Liu, L.: Physctrl: Generative physics for controllable and physics-grounded video generation. arXiv preprint arXiv:2509.20358 (2025)

[89] Wang, Y., Li, Y., Qian, G.G., Tulyakov, S., Fu, Y., Kag, A.: Diffusion-drf: Free, rich, and differentiable reward for video diffusion fine-tuning (2026), `https://arxiv.org/abs/2601.04153`

[90] Wang, Y., Qian, G.G., Li, Y., Kag, A., Fu, Y.: Token-level video reinforcement learning (2026), `https://arxiv.org/abs/2610.01973`

[91] Wang, Z., Yuan, Z., Wang, X., Chen, T., Xia, M., Luo, P., Shan, Y.: Motionctrl: A unified and flexible motion controller for video generation. arXiv preprint arXiv:2312.03641 (2023)

[92] Yang, S., Du, Y., Ghasemipour, S., Tompson, J., Kaelbling, L., Schuurmans, D., Abbeel, P.: Learning interactive real-world simulators. In: International Conference on Learning Representations (2024)

[93] Ye, S., Jang, J., Jeon, B., Joo, S., Yang, J., Peng, B., Mandlekar, A., Tan, R., Chao, Y.W., Lin, B.Y., et al.: Latent action pretraining from videos. In: International Conference on Learning Representations (2025)

[94] Yin, S., Wu, C., Liang, J., Shi, J., Li, H., Ming, G., Duan, N.: Dragnuwa: Fine-grained control in video generation by integrating text, image, and trajectory. arXiv preprint arXiv:2308.08089 (2023)

[95] Yu, T., Li, X., Shen, Y., Shen, Y., Nguyen, K.A., Juvekar, A.S., Lourentzou, I.: Silsa: Sliding-window slice latents for topology-preserving high-resolution 3d generation. arXiv preprint arXiv:2610.02201 (2026)

[96] Yu, T., Li, X., Shen, Y., Susladkar, O., Liu, Y., Zhou, X., Lourentzou, I.: Elsa3d: Elastic semantic anchoring for unified 3d understanding and generation. arXiv preprint arXiv:2607.06565 (2026)

[97] Zhang, H., Chai, W., He, S., Li, A., Fu, Y.: Dive-bench: Codec-style high-fps video understanding. In: European Conference on Computer Vision. pp. 418–435. Springer (2026)

[98] Zhang, H., Chen, M., He, S., Xu, Z., Shen, Y., Huang, Y., Lu, J., Li, Y., She, Y., Fu, Y.: A survey of physical ai: A history from chatgpt to world models and embodied agents. Preprints, June (2026)

[99] Zhang, H., Fu, Y.: Vqtoken: Neural discrete token representation learning for extreme token reduction in video large language models. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems (2025)

[100] Zhang, H., Li, Y., He, S., Nagarajan, T., Chen, M., Lu, J., Li, A., Fu, Y.: Thinkjepa: Empowering latent world models with large vision-language reasoning model. arXiv preprint arXiv:2603.22281 (2026)

[101] Zhang, H., Lu, Y., Wang, L., Li, Y., Chen, D., Xu, Y., Fu, Y.: Linkedout: Linking world knowledge representation out of video llm for next-generation video recommendation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7111–7121 (2026)

[102] Zhang, H., Xu, Y., Fu, Y.: Out-of-sight embodied agents: Multimodal tracking, sensor fusion, and trajectory forecasting. IEEE Transactions on Pattern Analysis and Machine Intelligence (2026)

[103] Zhang, H., Xu, Y., Lu, H., Shimizu, T., Fu, Y.: Layout sequence prediction from noisy mobile modality. In: Proceedings of the 31st ACM International Conference on Multimedia. pp. 3965–3974 (2023)

[104] Zhang, H., Xu, Y., Lu, H., Shimizu, T., Fu, Y.: Oostraj: Out-of-sight trajectory prediction with vision-positioning denoising. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14802–14811 (2024)

[105] Zhang, J., Jiang, M., Dai, N., Lu, T., Uzunoglu, A., Zhang, S., Wei, Y., Wang, J., Patel, V.M., Liang, P.P., Khashabi, D., Peng, C., Chellappa, R., Shu, T., Yuille, A., Du, Y., Chen, J.: World-in-World: World models in a closed-loop world. In: International Conference on Learning Representations (2026), arXiv:2510.18135

[106] Zhou, G., Pan, H., Lecun, Y., Pinto, L.: Dino-wm: World models on pre-trained visual features enable zero-shot planning. In: International Conference on Machine Learning. pp. 79115–79135. PMLR (2025)

[107] Zhou, G., Pan, H., Lecun, Y., Pinto, L.: Dino-wm: World models on pre-trained visual features enable zero-shot planning. In: International Conference on Machine Learning. pp. 79115–79135. PMLR (2025)

[108] Ziakas, C., Bar, A., Russo, A.: Grounding generated videos in feasible plans via world models. arXiv preprint arXiv:2602.01960 (2026)

A Additional Analysis of Action-Conditioned Predictor Generalization

A.1 Encoder Generality Does Not Guarantee Predictor Generalization

A JEPA-style latent world model contains two different components: a visual encoder and a transition predictor. Let E be a frozen JEPA encoder pretrained on a broad video distribution $\mathcal{D}_E(x)$. Let P_θ be an action-conditioned predictor trained on an embodied action-conditioned dataset $\mathcal{D}_{\mathrm{AC}}(h, a, h')$. The encoder learns a representation map

$$E : x \mapsto h,$$

whereas the action-conditioned predictor must learn a conditional transition

$$P_\theta : (h, a) \mapsto h'.$$

These are different generalization problems. Broad pretraining of E can provide useful visual representations, but the predictor still depends on the coverage of the joint latent-action distribution $\mathcal{D}_{\mathrm{AC}}(h, a)$. Therefore, encoder generalization over observations does not imply transition generalization over latent-action pairs.

In practice, $\mathcal{D}_{\mathrm{AC}}$ is determined by the embodied dataset used for predictor training. Its support is constrained by the robot embodiment, camera setup, action parameterization, task set, object distribution, and data collection policy. If the test distribution contains unseen action categories, new objects, new tasks, or new embodiments, then the predictor is evaluated on latent-action pairs that may be weakly covered or entirely unsupported by the training distribution. This is the sense in which the action-conditioned predictor becomes the bottleneck.

A.2 Distribution-Shift Bound

Let $z = (h, a)$ denote the latent state and planned action input, and let $y = h'$ denote the future latent target. For a predictor $P \in \mathcal{H}$ and a loss ℓ, define the risk

$$R_{\mathcal{D}}(P) = \mathbb{E}_{(z,y)\sim\mathcal{D}} \left[\ell(P(z), y)\right].$$

Let $\mathcal{D}_{\mathrm{tr}}$ and $\mathcal{D}_{\mathrm{te}}$ denote train and test distributions. Consider a loss-induced hypothesis discrepancy

$$\mathrm{Disc}_{\mathcal{H}}(\mathcal{D}_{\mathrm{tr}}, \mathcal{D}_{\mathrm{te}}) = \sup_{P,Q\in\mathcal{H}} \left|\mathbb{E}_{z\sim\mathcal{D}_{\mathrm{tr}}} \ell(P(z), Q(z)) - \mathbb{E}_{z\sim\mathcal{D}_{\mathrm{te}}} \ell(P(z), Q(z))\right|.$$

By standard domain-adaptation arguments, for any $P \in \mathcal{H}$,

$$R_{\mathrm{te}}(P) \le R_{\mathrm{tr}}(P) + \mathrm{Disc}_{\mathcal{H}}(\mathcal{D}_{\mathrm{tr}}, \mathcal{D}_{\mathrm{te}}) + \lambda^\star,$$

where

$$\lambda^\star = \min_{P\in\mathcal{H}} R_{\mathrm{tr}}(P) + R_{\mathrm{te}}(P).$$

This bound emphasizes that low training risk does not guarantee low test risk under a large latent-action distribution shift. A broadly pretrained encoder may reduce the difficulty of representing observations, but it does not remove the discrepancy in the conditional transition-learning problem.

A.3 Non-Identifiability Outside the Training Support

Let

$$\mathcal{S}_{\mathrm{tr}} = \mathrm{supp}(\mathcal{D}_{\mathrm{tr}}(z)).$$

Assume there exists a held-out test input $z^\star = (h^\star, a^\star)$ such that

$$z^\star \notin \mathcal{S}_{\mathrm{tr}}.$$

Then there exist two transition functions f_1 and f_2 satisfying

$$f_1(z) = f_2(z), \qquad \forall z \in \mathcal{S}_{\mathrm{tr}},$$

but

$$f_1(z^\star) \neq f_2(z^\star).$$

Both functions induce identical training observations and identical empirical risk, yet they disagree on the held-out region. Therefore, the transition at $z^\star$ is not identifiable from the action-conditioned

training data alone. Additional priors or broader support are required for reliable out-of-distribution transition prediction.

This non-identifiability result does not imply that generalization is impossible. Rather, it shows that a predictor trained only on a narrow embodied support cannot be guaranteed to infer the correct transition in unseen regions without additional inductive bias or external dynamics priors. JEPAGeneralist uses frozen controllable video world models as such an external prior.

A.4 Video World Models as Broader Motion Priors

A controllable video world model defines a video-space conditional distribution

$$\tilde{x}_{t+1:t+T} \sim G_{\mathrm{VWM}}(\cdot \mid r(a_{t+1:t+T})),$$

where $r(a)$ is a rendered visual control signal. The control signal may be a skeleton, trajectory, mask, reference, or other video-space condition. We do not interpret this interface as a universal robot action space. Instead, it is a broad visual-control manifold that can express diverse action-like motion cues.

After encoding generated rollouts with the frozen JEPA encoder,

$$\tilde{h}_{t+1:t+T} = E(\tilde{x}_{t+1:t+T}),$$

the generated rollouts induce latent motion references. Because the video world model is trained on broader video distributions than the action-conditioned predictor, these generated rollouts can provide motion priors outside the narrow support of the embodied training data. However, generated rollouts may not preserve the source content. Therefore, they should be used as motion-reward supervision rather than full future-latent targets.

B Additional Analysis of Generated Futures and Motion-Reward Distillation

B.1 Bias from Direct Generated-Latent Distillation

We provide a simple analysis showing why directly distilling generated futures can bias an action-conditioned latent predictor when generated content does not match the observed source context.

Assume a real future latent decomposes as

$$h^{\mathrm{real}} = c^{\mathrm{src}} + d(a) + \epsilon,$$

where c^{src} denotes source-conditioned content continuation, $d(a)$ denotes action-induced dynamics, and ϵ is residual noise. A generated future latent under the same action decomposes as

$$\tilde{h} = c^{\mathrm{gen}} + d(a) + \eta,$$

where c^{gen} may differ from c^{src} because a video world model may change the environment, object identity, visual appearance, or generation style.

Consider optimizing a predictor output y with a weighted combination of real and generated squared losses:

$$\min_y \|y - h^{\mathrm{real}}\|_2^2 + \lambda \|y - \tilde{h}\|_2^2.$$

The minimizer is obtained by setting the derivative to zero:

$$2(y - h^{\mathrm{real}}) + 2\lambda(y - \tilde{h}) = 0,$$

which gives

$$y^\star = \frac{h^{\mathrm{real}} + \lambda \tilde{h}}{1 + \lambda}.$$

Substituting the decompositions of h^{real} and $\tilde{h}$ yields

$$y^\star = c^{\mathrm{src}} + d(a) + \frac{\lambda}{1+\lambda}(c^{\mathrm{gen}} - c^{\mathrm{src}}) + \frac{\epsilon + \lambda\eta}{1+\lambda}.$$

The term

$$\frac{\lambda}{1+\lambda}(c^{\mathrm{gen}} - c^{\mathrm{src}})$$

is a content-bias term induced by generated futures. It is unrelated to the desired action-induced dynamics and can corrupt source-conditioned future prediction. Thus, generated futures should not be used as full latent targets when their content is not aligned with the observed source context.

B.2 Why Patch-Level Motion Rewards Avoid the Content Bias

In JEPAGeneralist, the full predicted future latent is anchored by real data:

$$\hat{h} = F_\psi(c, d), \qquad \mathcal{L}_{\text{real}} = D(\hat{h}, h^{\text{real}}).$$

Generated rollouts do not directly supervise $\hat{h}$ as absolute future targets. Instead, we use only their temporal changes as offline VWM motion rewards. Given the predicted future latent $\hat{h}_{t+1:t+T}$ and the generated future latent $\tilde{h}_{t+1:t+T}$, we compute

$$\Delta\hat{h}_{t+i} = \hat{h}_{t+i+1} - \hat{h}_{t+i}, \qquad \Delta\tilde{h}_{t+i} = \tilde{h}_{t+i+1} - \tilde{h}_{t+i}, \quad i = 1, \ldots, T-1.$$

The motion-reward distillation loss is

$$\mathcal{L}_{\text{MR}} = D\Big(\Delta\hat{h}_{t+1:t+T-1}, \Delta\tilde{h}_{t+1:t+T-1}\Big).$$

This objective encourages the predicted temporal dynamics to follow same-action generated motion while preventing source-mismatched generated content from becoming an absolute latent target. It therefore preserves the real-data anchor for source-conditioned content while transferring useful temporal-dynamics priors from video world models.

B.3 Offline VWM Motion-Reward Bank

The VWM motion-reward bank is constructed offline. For each training sample, we render the planned action proxy into a visual control signal,

$$u_{t+1:t+T} = r(a_{t+1:t+T}),$$

and use a frozen controllable video world model to generate a same-action future rollout:

$$\tilde{x}_{t+1:t+T} \sim G_{\text{VWM}}(\cdot \mid u_{t+1:t+T}).$$

The generated rollout is then encoded by the frozen JEPA encoder:

$$\tilde{h}_{t+1:t+T} = E(\tilde{x}_{t+1:t+T}).$$

For each sample, the bank stores

$$\Big(h_{\le t}, h_{t+1:t+T}, \tilde{h}_{t+1:t+T}, a_{t+1:t+T}\Big),$$

together with metadata needed for alignment and evaluation. The generated latent is stored at patch level, so the motion reward can be computed over token-level temporal deltas. During predictor training, no video generation, video decoding, or generated-video JEPA encoding is performed inside the optimization loop.

C Supplementary Details for Experimental Variants

This section summarizes the experimental variants used in the main paper. We include these definitions to clarify the role of each control without introducing additional performance claims beyond the main results.

Basic. The Basic predictor denotes the standard action-conditioned latent predictor. It directly maps observed past latents and action inputs to future JEPA latents without content–dynamics factorization or generated supervision.

GenAbs. GenAbs adds absolute generated-latent supervision from VWM-generated rollouts. It tests the naive alternative of using generated future latents as direct future targets. As discussed in Appendix B.1, this can introduce source-mismatched content bias when generated scenes, objects, or appearances differ from the real future.

Factorized. The factorized predictor separates future latent prediction into a source-conditioned content branch and an action-induced dynamics branch. This decomposition reduces interference between static content continuation and action-dependent temporal change.

Cross-Attn. Cross-Attn denotes the content-anchored cross-attention fusion module used in the main calibrated variant of JEPAGeneralist. It allows content tokens to attend to action-induced dynamics cues, injecting motion while preserving the calibrated JEPA latent manifold.

GatedAdd. GatedAdd denotes a gated additive fusion variant studied in ablations. It injects the dynamics branch through a learned residual gate. This variant can improve action recoverability and action sensitivity, but may trade off latent calibration.

Aligned GenΔ. Aligned GenΔ is our offline VWM motion reward. It uses the temporal deltas of the clip-aligned generated future latent as supervision:

$$\Delta\tilde{h}_{t+i} = \tilde{h}_{t+i+1} - \tilde{h}_{t+i}.$$

This transfers same-action motion information without treating generated content as an absolute target.

Shuffled GenΔ. Shuffled GenΔ is a negative control. It uses the same generated temporal-delta targets but randomly mismatches them with different source clips. This tests whether the benefit of Aligned GenΔ comes from clip-aligned generated motion rather than generic temporal regularization.

Real-Δ Aux. Real-Δ Aux adds a lightweight temporal smoothing term to the content loss using deltas computed from real future latents. It tests whether the gains from Aligned GenΔ can be explained solely by a generic temporal-delta auxiliary loss.

JEPAGeneralist (Cross-Attn). This is the main calibrated variant of JEPAGeneralist. It uses content–dynamics factorization, Cross-Attn fusion, and Aligned GenΔ with auxiliary weight 0.1, corresponding to $\lambda_{\mathrm{MR}} = 0.1$.

JEPAGeneralist-GatedAdd. This variant replaces Cross-Attn fusion with GatedAdd fusion while retaining Aligned GenΔ. It is included to analyze the trade-off between latent calibration and action recoverability.

D Reproducibility Details

Frozen representations. All latent prediction experiments operate in frozen representation spaces. The JEPA encoder is not finetuned during predictor training. Generated videos are also encoded offline, and the resulting latent tensors are cached before predictor optimization.

Offline generation. The controllable video world model is used only for offline bank construction. It is not queried during predictor training or inference. This design amortizes the cost of pixel-space generation and ensures that all predictor variants can be compared using the same cached real and generated latents.

Training and checkpoint selection. The predictor is first trained with a real-only warmup before enabling the VWM motion reward. The main configuration uses $\lambda_{\mathrm{MR}} = 0.1$. Checkpoints are selected using held-out real future prediction metrics rather than generated-video scores, ensuring that generated supervision does not dominate the real-data objective.

Evaluation. The main paper reports latent prediction, action sensitivity, action recoverability, and physical contact readout metrics. For cross-latent-space comparisons, raw MSE and cosine similarity are not compared across incompatible feature spaces; instead, we emphasize action recoverability through backbone-specific inverse dynamics probes.

D.1 Training Configuration, Data Banks, and Compute Resources

Accounting scope and data splits. Table 8 reports the data splits and bank sizes recorded for the controlled audits. The EgoDex counts and predictor costs below concern the fixed 483-trajectory audit protocol used in Section 4.4. The audit covers all four held-out categories, whereas the original

per-subtask table displays three. Its fixed manifest is separate from the broader 111-task official-test evaluation in Section 4.3. The 483 held-out trajectories comprise lamp assembly/disassembly (62), square-table assembly/disassembly (163), basic fold (157), and round-table insertion/removal (101), with no train/validation task or file overlap. BAIR uses real cached latents, and Physion uses real-latent contact readout across eight scenarios; neither uses a generated/VWM bank.

Table 8: Data and offline-bank accounting for the controlled audits. Bank entries give raw candidates $\rightarrow$ retained candidates, followed by serialized cache size. MiB denotes disk storage, not GPU memory.

Dataset	Train	Val.	Test	Generated bank
EgoDex	1,367	150	483	Train: $10{,}936 \rightarrow 5{,}468$ (185 MiB); evaluation: $3{,}864 \rightarrow 1{,}932$ (66 MiB)
BAIR	2,000	500	256	None
Physion	2,000	400	1,200	None

Offline bank construction. We use the provider-released `Wan-AI/Wan2.1-VACE-1.3B` as the fixed teacher, without EgoDex finetuning. Generation uses rendered skeleton control with no source-reference image, a resolution of 480×832, 17 future frames, and 12 inference steps. For each source/control pair, eight stochastic candidates are generated and four are retained according to V-JEPA2 consistency, control-rendering validity, and generation validity. The four retained candidates describe bank capacity; the controlled source comparison uses one retained candidate per source/control in each arm. Accepted RGB rollouts are encoded once by frozen V-JEPA2. Predictor optimization reads cached representations. Neither optimization nor inference invokes the video generator, decodes generated videos, or encodes generated RGB; inference still requires encoding the observed real history with the frozen encoder.

Predictor training configuration. The controlled EgoDex audits use 17 observed frames and 17 future frames. Optimization uses AdamW with learning rate 10^{-4} and weight decay 10^{-4}, batch size 16 per GPU on four GPUs (global batch size 64), and 200 configured epochs ending at 17,200 optimizer updates. The first 860 updates are real-only warmup. The main configuration uses $\lambda_{\mathrm{MR}} = 0.1$, whereas the separate source-control audit in Section 4.4 uses a validation-selected weight of 0.05 for GenΔ and a training-statistic-selected weight of 0.10 for Real-Δ. Runs use seeds 42, 43, and 44, with checkpoints selected by validation real-future MSE.

Measured predictor compute. Runs used RTX PRO 6000 Blackwell and B200 GPUs. Table 9 reports aggregate predictor GPU-hours, not per-run wall-clock times or end-to-end training costs. The per-configuration entries are totals across three seeds. These measurements document consumed resources on the recorded hardware and are not hardware-normalized speed comparisons. Audit totals and per-configuration totals are separate accounting views and should not be added together.

Table 9: Measured predictor compute for the controlled EgoDex audits. Offline video generation, generated-video encoding, and bank filtering are excluded. The Slurm entry is an alternative accounting of the architecture/fusion audit including preflight, not an additional cost.

Accounting scope	GPU-hours
Architecture/fusion audit (26 runs; predictor optimization)	178.11
Architecture/fusion audit (Slurm accounting including preflight)	178.86
Source audit (12 runs; predictor optimization)	113.00
Standard real-only (three-seed total)	19.57
Factorized real-only (three-seed total)	25.00
Real-Δ (three-seed total)	24.47
GenAbs (three-seed total)	11.90
GatedAdd (three-seed total)	20.83
Complete default (three-seed total)	27.00

Accounting limitations. Historical RGB-generation wall-clock time was not preserved in the recovered logs. Separate GPU-hour measurements for offline encoding and filtering are not available

in the retained accounting used here. The retained records also do not provide per-run peak GPU-memory usage or complete per-experiment and full-project compute accounting. We therefore report the measured predictor costs, exact source/candidate counts, cache storage, and generation configuration without estimating missing offline costs. The operational claim is that bank construction is one-time and amortized while predictor optimization and inference are VWM-free; the reported GPU-hours do not establish end-to-end compute savings.

E Experimental Details

This section provides detailed dataset protocols, baseline definitions, and evaluation metrics for the experiments in Section 4. Experimental variant definitions, including the two final JEPAGeneralist configurations, are consolidated in Appendix C.

E.1 Datasets and Protocols

EgoDex. EgoDex [43] is our primary benchmark for held-out cross-task generalization. Models are trained on seen embodied action/task categories and evaluated on three disjoint held-out subtasks: `assemble square table`, `fold`, and `insert round table`. This setting directly tests whether an action-conditioned latent world model can generalize beyond the action and task distributions observed during training. Since the held-out subtasks have imbalanced sample counts, the main paper reports both aggregate and per-subtask action recoverability.

BAIR Robot Pushing. BAIR Robot Pushing [33] is used as a complementary future-frame latent-prediction benchmark. All BAIR methods are evaluated in the same V-JEPA latent space. Given past latent tokens $\mathbf{h}_{\text{past}}$ and an action sequence $\mathbf{a}$, the model predicts future latent tokens

$$\hat{\mathbf{h}}_{1:T} = f(\mathbf{h}_{\text{past}}, \mathbf{a}),$$

which are compared against ground-truth future latents $\mathbf{h}_{1:T}$ extracted by the same frozen V-JEPA encoder. Since these metrics are computed directly in latent space, they are comparable only among models using the same latent backbone.

Physion. Physion [10] is used to evaluate whether predicted future latents preserve physically relevant contact information. We train a token-level Transformer readout for physical contact prediction. The readout input consists of observed V-JEPA tokens and predicted future V-JEPA tokens. We use 2,000 training samples, 400 validation samples, and 1,200 test samples. The protocol uses $T_{\text{past}} = 32$ and $T_{\text{future}} = 32$, with no action input. Each sample contains 256 observed tokens and 128 predicted future tokens. The readout is trained for 50 epochs and selected by validation readout performance.

E.2 Baselines

EgoDex baselines. On EgoDex, we compare JEPAGeneralist with frozen-backbone latent predictor baselines. **V-JEPA2** [5] denotes a frozen V-JEPA2 latent backbone with a shallow action-conditioned MLP predictor. **V-JEPA2 AC** [5] denotes a frozen V-JEPA2 latent backbone with a standard action-conditioned Transformer predictor. **DINOv2** [72] denotes a frozen DINOv2 latent backbone with a shallow action-conditioned MLP predictor. **DINO-WM** [107] denotes a DINO-WM-style action-conditioned Transformer predictor trained on frozen DINOv2 features. All baselines use cached frozen features and are trained only in latent space, without pixel-level finetuning or test-time optimization. Since V-JEPA2 and DINOv2 features have different scales and feature distributions, raw latent MSE and cosine similarity are comparable only within the same latent space. For cross-backbone comparison, we emphasize action recoverability through backbone-specific inverse dynamics probes.

BAIR baselines. On BAIR, we compare against V-JEPA-space latent predictors, including a shallow V-JEPA MLP baseline and a standard V-JEPA Predictor. The JEPAGeneralist + Aux variant adds a lightweight smoothing auxiliary term on the content loss.

Physion baselines. On Physion, we compare against DINO-WM, V-JEPA observed-only latents, and a V-JEPA AC Predictor. This evaluates whether predicted future latents improve downstream physical contact readout beyond observed-only and standard predictor baselines.

E.3 Evaluation Metrics

Latent prediction. For latent prediction, we report mean squared error (MSE) between predicted and ground-truth future latents:

$$\mathrm{MSE} = \frac{1}{TPD} \sum_{t=1}^{T} \sum_{p=1}^{P} \sum_{d=1}^{D} \left(\hat{\mathbf{h}}_{t,p,d} - \mathbf{h}_{t,p,d} \right)^2 ,$$

where T is the number of future frames, P is the number of latent tokens per frame, and D is the latent dimension. We also report cosine similarity after flattening all future-frame latent tokens:

$$\mathrm{Cos} = \frac{\left\langle \mathrm{vec}(\hat{\mathbf{h}}_{1:T}), \mathrm{vec}(\mathbf{h}_{1:T}) \right\rangle}{\left\| \mathrm{vec}(\hat{\mathbf{h}}_{1:T}) \right\|_2 \left\| \mathrm{vec}(\mathbf{h}_{1:T}) \right\|_2}.$$

Retrieval accuracy. Retrieval accuracy evaluates whether the predicted future representation retrieves its paired ground-truth future representation from the test pool. For each predicted future latent, all ground-truth future latents in the test set are ranked by similarity. R@1 is correct if the paired future latent is ranked first, and R@5 is correct if it appears among the top five retrieved candidates.

Temporal latent-delta error. We report temporal latent-delta error (ΔErr) in the component ablation. It compares predicted and real future temporal changes:

$$\Delta\mathrm{Err} = D\Big(\Delta\hat{\mathbf{h}}_{1:T-1}, \Delta\mathbf{h}_{1:T-1} \Big) ,$$

where

$$\Delta\hat{\mathbf{h}}_t = \hat{\mathbf{h}}_{t+1} - \hat{\mathbf{h}}_t, \qquad \Delta\mathbf{h}_t = \mathbf{h}_{t+1} - \mathbf{h}_t.$$

Lower ΔErr indicates better temporal-dynamics prediction in the target latent space.

Action sensitivity. Shuffle sensitivity replaces the correct action sequence with an action sequence from another sample:

$$\mathrm{ShuffleSens} = \mathrm{MSE}\big(f(\mathbf{h}_{\mathrm{past}}, \mathbf{a}_{\mathrm{shuffle}}), \mathbf{h}_{1:T} \big) - \mathrm{MSE}\big(f(\mathbf{h}_{\mathrm{past}}, \mathbf{a}), \mathbf{h}_{1:T} \big).$$

Zero sensitivity replaces the action sequence with zeros:

$$\mathrm{ZeroSens} = \mathrm{MSE}\big(f(\mathbf{h}_{\mathrm{past}}, \mathbf{0}), \mathbf{h}_{1:T} \big) - \mathrm{MSE}\big(f(\mathbf{h}_{\mathrm{past}}, \mathbf{a}), \mathbf{h}_{1:T} \big).$$

Larger values indicate that corrupting or removing the action input causes larger degradation, suggesting stronger use of action conditioning.

Action recoverability. For action recoverability on EgoDex, we use a schema-conditioned inverse dynamics model (IDM) probe. Given a predicted action trajectory $\hat{\mathbf{x}}_{1:T}$ and the ground-truth trajectory $\mathbf{x}_{1:T}$, ADE measures the average Euclidean distance over future timesteps:

$$\mathrm{ADE} = \frac{1}{T} \sum_{t=1}^{T} \left\| \hat{\mathbf{x}}_t - \mathbf{x}_t \right\|_2 .$$

FDE measures the endpoint error at the final future timestep:

$$\mathrm{FDE} = \left\| \hat{\mathbf{x}}_T - \mathbf{x}_T \right\|_2 .$$

PCK@0.05 measures the fraction of predicted action points whose distance to the ground truth is below a normalized threshold of 0.05:

$$\mathrm{PCK@0.05} = \frac{1}{T} \sum_{t=1}^{T} \mathbb{I}\left[\left\| \hat{\mathbf{x}}_t - \mathbf{x}_t \right\|_2 < 0.05 \right] .$$

Lower ADE/FDE and higher PCK indicate better recovery of action-relevant dynamics from predicted future latents.

Physion contact readout. For Physion contact readout, we report Contact Accuracy, Balanced Accuracy, and ROC-AUC. Given a binary contact label $y_i \in \{0, 1\}$ and a predicted contact probability p_i, Contact Accuracy is

$$\mathrm{Acc} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}\left[\mathbb{I}(p_i > 0.5) = y_i\right].$$

Balanced Accuracy averages the true positive rate and true negative rate:

$$\mathrm{BalancedAcc} = \frac{1}{2}\left(\frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} + \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}\right).$$

ROC-AUC measures the threshold-independent ranking quality of predicted contact scores. Higher Contact Accuracy, Balanced Accuracy, and ROC-AUC indicate better physical contact readout from the predicted future latents.

E.4 Qualitative Visualization Protocol

For qualitative action-trajectory visualization, we decode future hand trajectories from predicted future latents using the same schema-conditioned IDM probe used in the quantitative action-recoverability evaluation. The decoded 3D joints are overlaid on a reference frame. The color gradient indicates temporal progression, from red at the trajectory start to blue at the final endpoint. This visualization allows us to inspect whether predicted future latents preserve temporally coherent and action-consistent hand motion under held-out generalization.

E.5 Implementation and Reproducibility Notes

All latent prediction experiments operate in frozen representation spaces. The JEPA encoder is not finetuned during predictor training. Generated videos are encoded offline, and the resulting latent tensors are cached before predictor optimization. The controllable video world model is not queried during predictor training or inference. This design amortizes the cost of pixel-space generation and ensures that predictor variants can be compared using the same cached real and generated latents.

The predictor is trained with a real-only warmup before enabling the VWM motion reward. The main configuration uses $\lambda_{\mathrm{MR}} = 0.1$. Checkpoints are selected using held-out real future prediction metrics rather than generated-video scores, ensuring that generated supervision does not dominate the real-data objective. For cross-latent-space comparisons, raw MSE and cosine similarity are not compared across incompatible feature spaces; instead, we emphasize action recoverability through backbone-specific IDM probes.

F Broader Impact

This work studies generalist action-conditioned latent world models for video understanding, embodied prediction, and robotics-oriented representation learning. Potential positive impacts include improving the efficiency and generalization of latent world models, reducing the need for expensive real-world data collection, and enabling better evaluation of action-conditioned prediction under held-out categories. Because the method operates in latent space and uses video generation models only offline to provide motion-reward supervision, it does not directly release a general-purpose video generator or a deployed decision-making system.

Potential risks are mainly indirect. Improved motion prediction and video-world-model techniques could eventually be incorporated into systems for surveillance, manipulation, or synthetic media generation. We mitigate these risks by focusing on benchmark evaluation, latent prediction, and reproducible research artifacts rather than deployment or unrestricted generative applications. Any released code or intermediate artifacts are intended for research use and should be used consistently with the licenses and terms of the underlying datasets and pretrained models.

G Existing Assets and Licenses

Our experiments use existing public datasets, pretrained encoders, controllable video world models, and baseline implementations. We cite the original papers and repositories for all datasets, models,

and code assets used in this work. All assets are used for research purposes and according to their respective licenses and terms of use. When releasing code or derived intermediate artifacts, we will provide documentation describing the source assets, preprocessing steps, and intended research use.

H Limitations

Although our experiments provide evidence of task-level generalization, substantial work remains before scaling JEPAGeneralist into a broadly general-purpose latent world model. The current evaluation does not establish scaling behavior with increasing model capacity, training data, or teacher-generation budget, nor transfer across heterogeneous embodiments and action spaces. Offline generation preserves efficient inference but still incurs bank-construction and filtering costs at larger scales. Moreover, temporal differencing does not correct physically inconsistent or control-ignoring teacher motion, which can propagate through auxiliary supervision despite real-future anchoring. Broader training coverage, reliable control alignment, and systematic scaling studies are therefore needed to realize the potential of this approach.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the paper's main contributions: a generalist action-conditioned JEPA-style latent world model, an offline video-world-model generated temporal-delta supervision strategy, and evaluations under held-out action/task categories. The claims are supported by the experimental comparisons, ablations, and analysis presented in the main paper and appendix.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section H discusses the unestablished scaling behavior with model capacity, data, and teacher-generation budget; transfer across heterogeneous embodiments and action spaces; offline bank-construction and filtering costs; and the risk of propagating physically inconsistent or control-ignoring teacher motion through auxiliary supervision.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper does not present formal theoretical theorems or proofs. The risk decomposition and latent factorization discussion are used as conceptual motivation for the proposed method rather than as formal theoretical results.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper describes the model architecture, training objectives, generated-delta teacher bank construction, datasets, evaluation protocols, and main hyperparameters needed to reproduce the experimental results. Additional implementation details, preprocessing descriptions, and evaluation settings are provided in the supplemental material.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The paper provides detailed methodology, training objectives, dataset preparation, evaluation protocols, and hyperparameters needed to reproduce the main results. The code and scripts are currently undergoing administrative approval for public release, and we plan to open-source the reproducibility package upon acceptance.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The main paper and appendix specify the datasets, held-out category protocol, model variants, loss terms, warmup schedule, key hyperparameters, optimizer settings, and evaluation metrics. The appendix provides additional details for reproducing the training and test settings.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

Justification: The current submission emphasizes controlled comparisons under fixed train/test splits, consistent evaluation protocols, and ablations across multiple datasets and metrics. While full multi-seed error bars are computationally expensive for latent world-model training with generated-delta banks, the reported results are supported by consistent trends across datasets, metrics, and controlled model variants.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

Justification: Appendix D.1 reports the controlled audits' data splits, generated-bank counts and storage, teacher configuration, predictor training configuration, GPU types and count, and measured aggregate predictor GPU-hours. However, historical offline-generation time, separate encoding/filtering costs, per-run peak GPU-memory usage, and complete per-experiment and full-project compute accounting are not available in the retained records used here. We explicitly distinguish measured predictor costs from unmeasured offline bank construction rather than claim complete resource accounting.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: The research follows the NeurIPS Code of Ethics. It uses publicly available datasets and pretrained models under their respective terms, focuses on latent world-model learning and evaluation, and does not involve collecting new human-subject data or deploying a system that makes decisions about individuals.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The paper discusses broader impacts in the appendix. Potential positive impacts include more efficient and generalizable latent world models for embodied AI, robotics, and video understanding, while potential risks include misuse of improved video or motion prediction systems; the paper also emphasizes that the proposed method is evaluated as an offline research model rather than a deployed decision-making system.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out

that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release a high-risk pretrained language model, image generator, or newly scraped dataset. The released research artifacts, if any, are intended to reproduce latent world-model training and evaluation rather than to provide a general-purpose generation system.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper cites the original datasets, pretrained encoders, controllable video world models, and relevant baseline methods used in the experiments. The supplemental material summarizes the asset sources and states that all datasets and models are used according to their respective licenses and terms of use.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The paper introduces code and generated intermediate artifacts for constructing and using the offline generated-delta teacher bank. The supplemental material documents the data format, preprocessing pipeline, training commands, evaluation scripts, and intended research use.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing experiments or newly collected human-subject data. All experiments are conducted using existing datasets, pretrained models, and offline evaluation protocols.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not conduct new human-subject research or crowdsourcing. Therefore, IRB approval or equivalent review is not applicable to the experiments reported in this work.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method does not rely on LLMs as an important, original, or non-standard component. Any use of language models for writing, editing, or formatting assistance does not affect the methodology, experiments, scientific claims, or originality of the research.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.