DELTA: DUAL-STREAM ENCODING WITH LATENT TEMPORAL ANTICIPATION FOR FROZEN VISUAL WORLD MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Frozen vision backbones such as V-JEPA 2.1 provide strong visual features for manipulation, but not an action-conditioned model of dynamics. Existing world models introduce that dynamics under different modeling assumptions. Predictors defined on frozen features take a short temporal window and regress the next embedding, so they maintain no state beyond that window and do not cast the prediction as an update of the observed frame. Predictors that retain a longer history instead advance an image code or a learned latent, and therefore depart from the frozen representation. We propose DELTA, a reconstruction-free world model on frozen V-JEPA 2.1 (ViT-B/384) features. A dual-stream latent carries state across time and is updated by an action-conditioned causal transition. A residual head writes the resulting short-horizon change onto the observed tokens. Actions enter only as conditioning. On manipulation demonstrations, DELTA lowers multi-step feature-prediction error relative to these baselines.

1 INTRODUCTION

Robotic manipulation rarely reduces to reacting to the current frame alone. Choosing an action often requires imagining how the scene will evolve under that action—reaching, pushing, or regrasping all depend on short-horizon contact dynamics as well as longer-horizon consequences. A natural abstraction is therefore a *forward model* $p(\text{future} \mid \text{observation}, a)$ that can be rolled out under candidate action sequences for prediction or planning.

Classical pixel-space world models learn such dynamics by reconstructing images, which is computationally heavy and can entangle control-relevant structure with appearance. A complementary line of work instead models dynamics in a learned visual representation, where pretrained encoders (e.g., JEPA- or DINO-style backbones) already provide strong features. This makes a practical recipe attractive: *freeze* a large vision backbone, and train only a lightweight, action-conditioned predictor on top.

This paper studies that recipe in the reconstruction-free setting. We build an action-conditioned world model on frozen V-JEPA 2.1 features, train it on robotic manipulation demonstrations, and evaluate multi-step prediction quality—open-loop, replan, and CEM probes versus horizon—on frozen tokens. Even with a strong frozen encoder, however, predicting future features under actions is nontrivial: teacher-forced one-step errors can look small, while open-loop rollouts compound as the horizon grows (Figure 1a). Two common remedies are incomplete on their own (Figure 1b): residual predictors stay sharp early but drift, whereas decoding full token features from a compact latent can preserve memory yet start from a higher short-horizon error. These observations motivate a dual-timescale design.

DELTA. We propose **DELTA** (*Dual-stream Encoding with Latent Temporal Anticipation*): a reconstruction-free world model that couples a dual-stream latent state with a residual feature head. Frozen V-JEPA 2.1 features are encoded into a continuous latent Z_t; an action-conditioned causal temporal model updates $Z_t \to Z_{t+1}$; and predictions are formed as $\hat{F}_{t+1} = \mathrm{pool}(F) + \Delta(Z_{t+1})$, so short-horizon fidelity stays anchored on the frozen features while mid-to-long horizon dynamics are carried in Z_t. Actions enter only as conditioning (no behavior-cloning loss in the main setup).

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

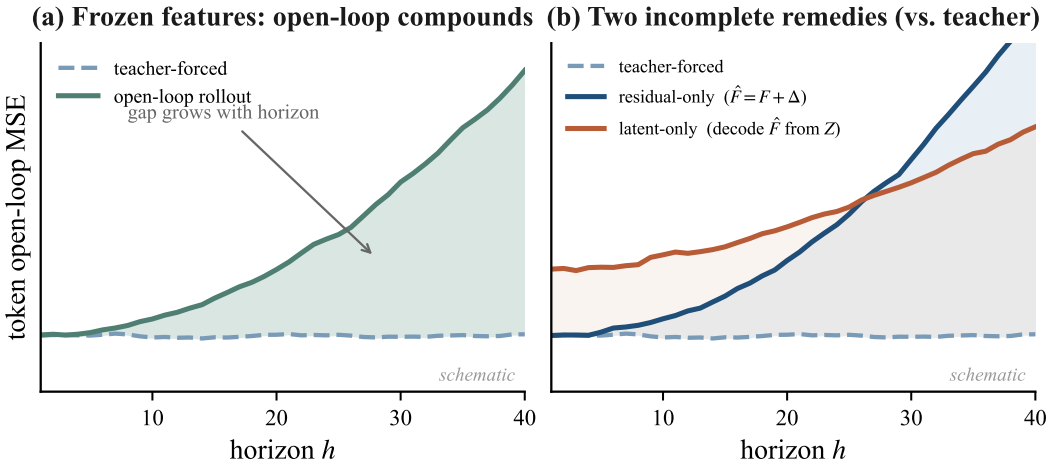

Figure 1: Schematic failure modes of feature-space world models. **(a)** On frozen features, open-loop token error compounds relative to teacher forcing. **(b)** Residual-only prediction is sharp early but drifts; latent-only decoding from Z starts worse and still rises. Neither timescale alone is enough.

Contributions.

- A reconstruction-free, action-conditioned world-model setting on **frozen V-JEPA 2.1**, with a shared feature-cache evaluation protocol for fair comparison.
- A dual-timescale architecture: dual-stream latent memory plus a residual token head $\hat{F}_{t+1} = \text{pool}(F) + \Delta(Z_{t+1})$.
- Empirical gains on robotic manipulation demonstrations in multi-step **token open-loop** and replan error, plus a CEM goal probe that ranks actions in the same feature space.

2 Related Work

Learning a forward model from pixels is an old idea, and a useful one. World Models, PlaNet, and Dreamer fit a latent dynamics model to images and then plan, or learn a policy, inside that model (Ha & Schmidhuber, 2018; Hafner et al., 2019; 2020). DreamerV2 and DreamerV3 carry the same latent-dynamics recipe to a wider set of domains (Hafner et al., 2021; 2025), including physical robots (Wu et al., 2023), transformer recurrent states (Chen et al., 2021), and MPC-style planning over learned dynamics (Hansen et al., 2022; 2024). Image reconstruction remains one of the training objectives in most of these systems, so the model predicts texture and lighting together with action-relevant change.

Another line of work learns the representation instead of the pixels. Vision transformers provide the patch-token backbone (Dosovitskiy et al., 2021; Vaswani et al., 2017), and masked autoencoders learn such tokens by reconstructing patches in images or video (He et al., 2022; Tong et al., 2022). DINO and DINOv2 train by self-distillation (with iBOT masking in DINOv2); the frozen features transfer to downstream tasks (Caron et al., 2021; Oquab et al., 2024). JEPA predicts missing content in representation space: I-JEPA from an image context (Assran et al., 2023), and V-JEPA from a video context (Bardes et al., 2024). V-JEPA 2 pretrains an action-free objective on over one million hours of internet video (Assran et al., 2025). The pretrained predictor does not take an action as input. The same paper then freezes the encoder and trains V-JEPA 2-AC, a block-causal transformer, on fewer than 62 hours of unlabeled robot video. V-JEPA 2-AC predicts the next-frame representation autoregressively, conditioned on actions and end-effector state, and plans with model-predictive control. Contrastive and masked world models likewise roll forward a learned latent rather than a frozen encoder field (Kipf et al., 2020; Seo et al., 2023).

DINO-WM puts an action-conditioned predictor on frozen DINOv2 patch features (Zhou et al., 2025). A vision transformer with frame-level causal attention reads a window of past embeddings and actions and predicts the next embedding itself, trained by teacher-forced latent consistency. The only history is that window: there is no latent kept apart from the pretrained features, and the prediction is not an update of the embedding just observed. A pixel decoder can be trained separately, for visualization; it is not part of the prediction loss. Navigation world models pursue a similar frozen-feature planning setting outdoors (Bar et al., 2025).

IRIS, TWM, and STORM keep a longer history, in a state trained to reconstruct images. IRIS turns each frame into discrete tokens for an autoregressive transformer (Micheli et al., 2023), with later tokenization refinements (Micheli et al., 2024). TWM runs a Transformer-XL over latent states, actions, and rewards (Robine et al., 2023). STORM represents each image by one categorical latent stepped by a transformer (Zhang et al., 2023). Diffusion world models and interactive video generators push the same idea toward richer pixel synthesis (Alonso et al., 2024; Bruce et al., 2024; Yang et al., 2024), while offline sequence and diffusion planners treat control as trajectory generation (Janner et al., 2021; 2022). The rolled state is therefore an image code or generative prior, not a frozen pretrained feature. DreamerPro removes the pixel decoder and predicts prototype assignments from the recurrent state (Deng et al., 2022). Our experiments use LIBERO Object demonstrations in a standard robot simulation stack (Liu et al., 2023; Zhu et al., 2020).

The comparison that follows therefore needs a shared observation space. We freeze one encoder, cache its tokens once, and score every method by how far those tokens drift under the same actions. That isolates the predictor: open-loop, replan, and a short CEM search all probe drift, recovery, and action ranking on those tokens, without a decoder. The architecture that targets that gap is introduced next.

3 METHOD

The following subsections write that recipe as equations. The encoder is the V-JEPA 2.1 ViT-B/384 checkpoint. V-JEPA 2.1 has no separate paper, so we cite V-JEPA 2 (Assran et al., 2025). Its weights are not updated. Only the modules after the encoder are trained. Features and actions are read from a fixed demonstration cache; the metric is a continuous token error on that cache. Notation uses time index t throughout: Z_t is the carried latent, a_t the action, and $\hat{F}_{t+1}$ the residual prediction. Figure 2 shows one step of this pipeline; the subsections below give the equations.

3.1 OBSERVATION FEATURES

Each training step is paired with two frozen feature tensors from the same encoder. The semantic tensor $F_{\mathrm{sem},t} \in \mathbb{R}^{N_s \times D}$ is the encoder output on the two frames ending at t, with $N_s{=}576$ and $D{=}768$. The dynamics tensor $F_{\mathrm{dyn},t} \in \mathbb{R}^{N_d \times D}$ is the encoder output on the four frames ending at t, with $N_d{=}1152$. The two are aligned to the same time index. These two tensors are the two streams. The semantic stream reads the two-frame window. The dynamics stream reads the four-frame window. Both are cached from LIBERO Object demonstrations (Liu et al., 2023) before training, and comparisons are run on this cache. Actions in the cache are standardized to zero mean and unit variance.

The prediction target is not the full patch grid. An adaptive average pool along the patch axis,

$$\mathrm{pool}_K(F) = \mathrm{AdaptiveAvgPool1d}_{N \to K}(F^\top)^\top \in \mathbb{R}^{K \times D}, \tag{1}$$

reduces either tensor to K tokens. In the main configuration $K{=}32$. The loss in section 3.5 is computed in this pooled space, which is also the space of the residual in equation 8.

3.2 DUAL-STREAM ENCODING

Pooling first would throw away the spatial layout the two windows still have. DELTA instead keeps K persistent tokens per stream, S_t and $D_t \in \mathbb{R}^{K \times d}$, and lets them read the features by cross-attention. The working dimension is $d{=}256$. At the start of an episode the tokens are learned constants; at later steps they are the latent produced for the previous time index. Under teacher forcing that latent is the encoding of the true observation. Under an open-loop rollout it is the

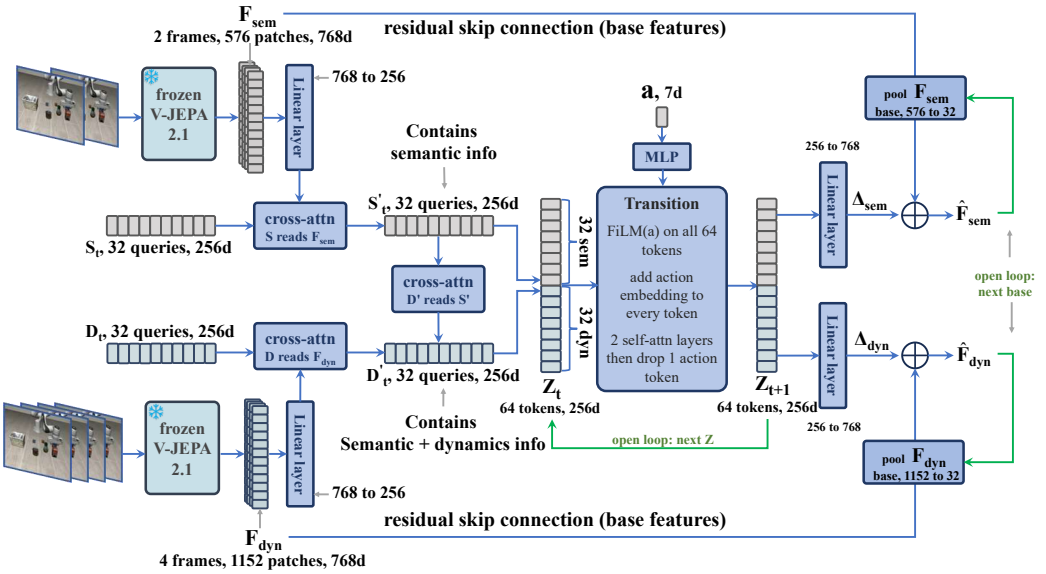

Figure 2: One step of DELTA. Left to right: a frozen V-JEPA 2.1 encoder produces semantic and dynamics features F_{sem} and F_{dyn}; linear layers and cross-attention read them into a dual-stream latent Z_t; an MLP embeds the action a; a transition block updates Z_t to Z_{t+1}; and linear residual heads predict Δ_{sem} and Δ_{dyn}, which are added to pooled base features to form $\hat{F}_{t+1}$. Solid arrows are one forward step. Dashed arrows are open-loop carry of Z_{t+1} and $\hat{F}_{t+1}$ to the next step.

model's own prediction. Linear maps Proj_s and Proj_d send the frozen features into $\mathbb{R}^d$. A pre-norm cross-attention block with 4 heads updates the queries and applies a feed-forward layer:

$$\begin{aligned} S'_t &= \text{CA}(S_t,\ \text{Proj}_s(F_{\text{sem},t})), \\ D'_t &= \text{CA}(D_t,\ \text{Proj}_d(F_{\text{dyn},t})). \end{aligned} \tag{2}$$

The dynamics stream then reads the semantic stream, and the two are joined:

$$\begin{aligned} D''_t &= \text{CA}(D'_t,\ S'_t), \\ Z_t &= \text{LN}\big([S'_t;\ D''_t]\big) \in \mathbb{R}^{2K\times d}. \end{aligned} \tag{3}$$

This second cross-attention is one way: the dynamics stream can use what the semantic stream just read, and not the other way around. Z_t is the state that has to carry time. A single-stream ablation drops the dynamics stream. It reads and predicts only the semantic stream. There is no dynamics reader and no coupling.

3.3 Action-conditioned transition

The transition does not see future actions. Given only (Z_t, a_t), it produces Z_{t+1}. The action is passed through a two-layer map with GELU and a layer norm, then scaled by a learned scalar α, initialized at 2,

$$e_a = \alpha \cdot \text{ActionEmbed}(a_t). \tag{4}$$

FiLM (Perez et al., 2018) then modulates every latent token. The FiLM layer is zero-initialized, so at the start of training the modulation is the identity:

$$(\gamma, \beta) = \text{split}(W_{\text{film}} e_a + b_{\text{film}}), \quad \tilde{Z}_t = Z_t \odot (1+\gamma) + \beta. \tag{5}$$

The product $(1+\gamma)$ is the amplification. The action scales every latent token. It is not only added to them. The same embedding is added to every token and concatenated as one extra token. L=2 pre-norm self-attention blocks, with 4 heads, mix the $2K$+1 tokens. Dropping the action token and applying a layer norm gives

$$Z^{(0)} = [\tilde{Z}_t + e_a;\ e_a], \qquad Z_{t+1} = \text{LN}\big(\text{SA}^{\circ L}(Z^{(0)})_{1:2K}\big). \tag{6}$$

Self-attention here is over tokens at one time index, not over a causal mask across time. Causality is the rollout: step $t{+}1$ is a function of Z_t and a_t only. There is no behavior-cloning head in the loss of section 3.5. A zero-action ablation replaces a_t with 0 and leaves the rest of the network unchanged.

3.4 Residual feature head

Decoding each next feature token from a d-dimensional latent token asks the head to synthesize the frozen feature in full. On a short horizon the current features are already a strong guess, which is the content of Figure 1b. The head therefore predicts an update. Two affine maps, with weights and biases zero-initialized so that training begins at a copy of the base, read the semantic and dynamics halves of Z_{t+1}, denoted $Z_{\mathrm{sem},t+1}$ and $Z_{\mathrm{dyn},t+1}$:

$$\Delta_{\mathrm{sem}} = W_s Z_{\mathrm{sem},t+1} + b_s, \quad \Delta_{\mathrm{dyn}} = W_d Z_{\mathrm{dyn},t+1} + b_d. \tag{7}$$

Each update is added to the pooled base,

$$\begin{aligned} \hat{F}_{\mathrm{sem},t+1} &= \mathrm{pool}_K(F_{\mathrm{sem,base}}) + \Delta_{\mathrm{sem}}, \\ \hat{F}_{\mathrm{dyn},t+1} &= \mathrm{pool}_K(F_{\mathrm{dyn,base}}) + \Delta_{\mathrm{dyn}}. \end{aligned} \tag{8}$$

If the base already has K tokens, pool_K is the identity. This is the $\hat{F}_{t+1} = \mathrm{pool}(F) + \Delta(Z_{t+1})$ of the introduction, written on both streams. The base depends on the protocol. Teacher forcing uses the observation at t as the base. An open-loop step uses the prediction from the previous step, starting from the true pooled features at the first index; these are the dashed arrows in Figure 2. The carried latent is not the same object in the two protocols. Under teacher forcing, the tokens passed forward are an encoding of the true next observation, read by the current tokens; the predicted latent is used only to form $\hat{F}$. An open-loop rollout encodes only the first observation and then feeds the predicted Z_{t+1} forward.

3.5 Objective

Training uses one distance on both streams. It is the sum of the two mean squared errors,

$$d(\hat{F}, F) = \mathrm{MSE}(\hat{F}_{\mathrm{sem}}, \mathrm{pool}_K(F_{\mathrm{sem}})) + \mathrm{MSE}(\hat{F}_{\mathrm{dyn}}, \mathrm{pool}_K(F_{\mathrm{dyn}})), \tag{9}$$

where MSE averages the squared error over the batch, tokens, and channels. Teacher forcing applies it one step ahead of the true observation,

$$\mathcal{L}_{\mathrm{pred}} = \mathbb{E}_t\big[d(\hat{F}_{t+1}, F_{t+1})\big], \tag{10}$$

with $F_{\mathrm{base}} = F_t$. That term can be small while an open-loop rollout, which no longer sees the true features, drifts. The second term is the same distance averaged over a k-step rollout that encodes only the start,

$$\mathcal{L}_{\mathrm{roll}} = \mathbb{E}\Big[\tfrac{1}{k}\sum_{i=0}^{k-1} d(\hat{F}_{i+1}, F_{i+1})\Big], \tag{11}$$

with $k{=}8$ in the main training run. A third term keeps those encoded latents from sitting still. Along a teacher-forced window, let $\overline{(Z_{t+1} - Z_t)^2}$ be the mean of the elementwise squared differences between consecutive encodings Z_t. Then

$$\mathcal{L}_{\mathrm{reg}} = \mathrm{ReLU}\big(\tau - \overline{(Z_{t+1} - Z_t)^2}\big), \tag{12}$$

with $\tau{=}0.015$. This term does not use the predicted latent. The training objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{pred}} + \lambda_{\mathrm{roll}}\mathcal{L}_{\mathrm{roll}} + \lambda_{\mathrm{reg}}\mathcal{L}_{\mathrm{reg}}. \tag{13}$$

$\lambda_{\mathrm{reg}}{=}1$ and $\lambda_{\mathrm{roll}}{=}1$. Latent alignment, behavior cloning, inverse dynamics, and a true-versus-false action margin are implemented and set to zero in the main experiment. They are not part of equation 13.

At evaluation the reported open-loop number is the mean squared error of the semantic prediction at horizon h, not the average of the steps before it. At a replan step the true observation is encoded from the initial tokens, the residual base is reset to the true pooled features, and the rollout continues. A CEM probe (Rubinstein, 1999) searches a short action sequence to reduce that same semantic error to a goal feature. It ranks candidate actions by that feature distance.

4 EXPERIMENTAL SETUP

The comparison uses one frozen feature cache and one metric. Every method is trained and scored on the same LIBERO Object demonstrations (Liu et al., 2023). The number is a feature-space dynamics error under shared actions.

4.1 DATA, METRIC, AND CHECKPOINTS

The cache is built from agent-view frames with the frozen V-JEPA 2.1 ViT-B/384 encoder. Semantic and dynamics features are the two streams of section 3.1, the two-frame window and the four-frame window, and the actions are standardized. A task-wise split with seed 0 gives 364 training episodes, 70 episodes used only to monitor validation loss, and 20 test episodes. Tables report means on the held-out test split. Checkpoints are not chosen on that reported evaluation.

The reported number is the mean squared error of the predicted semantic tokens, as in section 3.5. Open-loop starts from the true features at the first index and then rolls forward under demonstration actions, scoring compounding drift. Replanning every $r \in \{5, 10, 20\}$ steps puts the true observation back in place of the carried prediction, scoring recovery under resets. Horizons are 1, 5, 10, 20, and 40. The CEM probe searches a 10-step action sequence, with population 48, 5 iterations, and 8 elites, to reduce the same semantic error against the feature 10 steps ahead in the demonstration. It scores whether the model prefers demo-like actions over random ones in feature space—a planning probe for the world model.

DELTA is the model of section 3, with $d{=}256$, $K{=}32$, four heads, and $L{=}2$. DELTA, every ablation, and every trained baseline were trained to completion. The tables use those checkpoints.

4.2 BASELINES

All baselines consume the same pooled semantic tokens and the same actions. Under this shared feature-cache protocol we keep each model's architectural character and only adapt the interface to our tokens and metric; they differ in what they predict.

DINO-WM is the predictor of Zhou et al. (2025): a history of three frames, frame-level causal attention, the action embedding added on every token, and an absolute next-token output. There is no residual. Copy emits the current tokens and does not train. MLP-Δ maps the mean token and the action through a three-layer network and adds that one vector to every token. STORM keeps one latent of the current frame, steps it with a transformer conditioned on the action, and decodes a residual in token space (Zhang et al., 2023). TWM is a transformer over the current tokens and one action token, also trained as a residual (Robine et al., 2023). EMERALD is a masked transformer on the same continuous tokens, again with a residual output (Burchi & Timofte, 2025). A discrete codebook is left out: the metric is a continuous feature error. A recurrent state-space model that sees only a pooled vector is left out for the same reason.

5 RESULTS

On the held-out test split, DELTA is lowest or tied at every open-loop horizon in Table 1, and lowest under replan and CEM in Table 2. The largest gap against DINO-WM is at short horizon; removing the residual head in Table 3 is the ablation that hurts most. In-text feature errors use the same MSE $\times 10^2$ units as those tables; Figures 3 and 4 plot the unscaled MSE. The rest of this section reads those tables in order: open-loop first, then replan and CEM, then the ablations that assign each piece.

5.1 OPEN-LOOP AND REPLANNING

Figure 3 plots open-loop semantic token error against horizon. At one step, DELTA and TWM are tied at 0.20. From horizon 5 through 20, DELTA is alone at the bottom of the curve: 0.35, 0.45, and 0.92. Competing residual rows sit in the 0.40–0.70 band at $h{=}5$–10 and above 1.10 at $h{=}20$. At horizon 40, DELTA and MLP-Δ tie at 1.67, still below DINO-WM at 1.97.

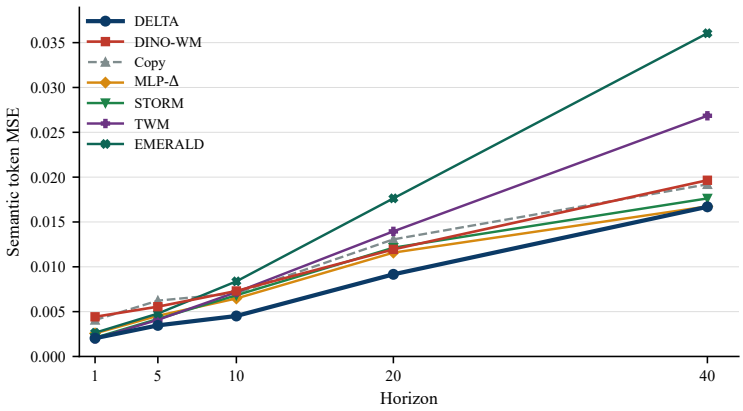

Figure 3: Open-loop semantic token error on the held-out test split. Markers are horizons 1, 5, 10, 20, and 40.

The curve shape is as important as the ranking. DINO-WM is the highest of the trained models at one step (0.44) and remains high at horizon 40. Copy has no dynamics: at one step it is 0.41, next to DINO-WM, and at horizon 40 the two are again close (1.92 and 1.97). Absolute next-token prediction therefore behaves like a near-copy once the horizon is long, whereas every residual row stays below that pair through the middle of the curve. The short-horizon gap is what the residual head is for; the mid-horizon spread is where a carried latent still separates the methods before the curves compress again at h=40. Between h=1 and h=10 the DELTA curve rises by only 0.25, while TWM rises by 0.52 and EMERALD by 0.58: early drift, not only the final h=40 value, is where the dual-stream residual pulls away. The same ordering already appears at h=5, so the gap is not an artifact of a single late horizon.

Table 1 makes the same ranking numerical and adds capacity. DELTA uses 5.52M predictor parameters, near DINO-WM (5.27M) and well below STORM (9.02M), so the open-loop gap is not a parameter blowout. TWM matches DELTA at h=1 then rises to 2.68 at h=40: a strong one-step residual does not guarantee a long rollout. EMERALD is already high by h=20 (1.76) and ends at 3.60, so a masked residual on the same tokens is not enough by itself. STORM stays competitive through h=20 (1.21) but does not undercut DELTA at any measured horizon despite the larger model. MLP-Δ is the closest residual peer at long horizon, yet it is already 0.20 worse than DELTA at h=10 (0.65 vs. 0.45). Capacity alone does not explain the ranking: smaller residual models still trail DELTA through the mid-horizon band.

Table 1: Main open-loop comparison on the held-out test split. Entries are MSE $\times 10^2$ (lower is bolded). Params are predictor parameters in millions.

Method	Params (M)	Open-loop semantic token MSE ($\times 10^2$)				
		h=1	h=5	h=10	h=20	h=40
DELTA	5.52	**0.20**	**0.35**	**0.45**	**0.92**	**1.67**
DINO-WM	5.27	0.44	0.56	0.73	1.19	1.97
Copy	0	0.41	0.62	0.70	1.31	1.92
MLP-Δ	3.68	0.26	0.45	0.65	1.16	**1.67**
STORM	9.02	0.21	0.42	0.68	1.21	1.76
TWM	3.63	**0.20**	0.41	0.72	1.40	2.68
EMERALD	3.10	0.26	0.48	0.84	1.76	3.60

Restoring the true observation does not change that open-loop order. Figure 4 shows three replan intervals on the same episodes: before the first restoration the curve matches open-loop, and each reset pulls every method down before error grows again. As r grows from 5 to 20, every method rises, but DELTA stays lowest at each setting in Table 2. The ranking is therefore stable under both

rare and frequent restoration, not an artifact of one reset schedule. Correcting the observation helps every method, yet DELTA keeps the lowest error after each reset as well as between them.

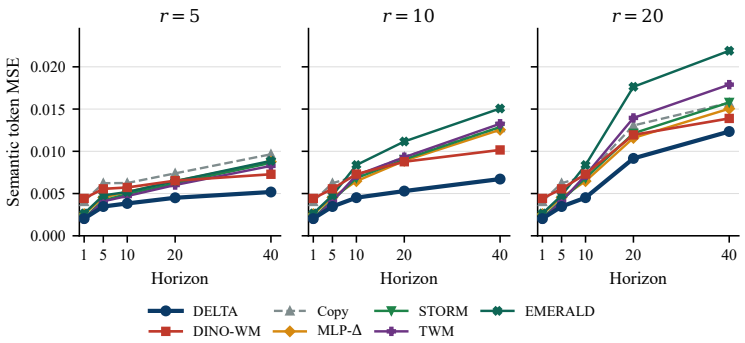

Figure 4: Replan curves: true observation restored every r steps. Panels share one scale; before the first restoration the curve matches open-loop.

Table 2: Replan error at h=40 and horizon-10 error under demonstration, random, and CEM actions. Entries are MSE $\times 10^2$ (lower is bolded). CEM ranks actions in feature space.

Method	Replan error at h=40 ($\times 10^2$)			Error at h=10 ($\times 10^2$)		
	r=5	r=10	r=20	Demo	Random	CEM
DELTA	**0.52**	**0.67**	**1.23**	**0.45**	**0.62**	**0.49**
DINO-WM	0.73	1.02	1.39	0.73	0.75	0.71
Copy	0.96	1.28	1.57	0.70	0.70	0.70
MLP-Δ	0.88	1.25	1.50	0.65	0.66	0.65
STORM	0.86	1.29	1.58	0.68	0.68	0.68
TWM	0.83	1.33	1.79	0.72	0.72	0.71
EMERALD	0.89	1.51	2.19	0.84	0.84	0.80

Reading Table 2 by columns makes the replan gap concrete. At r=5, DELTA (0.52) sits well below the residual cluster (0.83–0.89) and below DINO-WM (0.73). At r=10 the gap widens: DELTA is 0.67 while every other trained row is above 1.00, and EMERALD already reaches 1.51. At r=20 the ordering is unchanged but the absolute level is closer to open-loop h=40, as expected when restorations are rare. Under sparse restoration the panel slopes again peel EMERALD and TWM upward, echoing their open-loop blowup past h=20; a history window helps once the frame is corrected, but it does not erase the cost of predicting the next embedding in full.

The right block of the same table changes the actions instead of the observations. Under demonstration, random, and CEM actions at horizon 10, DELTA remains lowest (0.49 under CEM). MLP-Δ is next (0.65); DINO-WM ties TWM at 0.71. Copy is flat across the three columns because it never reads the action. Random actions raise DELTA from 0.45 (demo) to 0.62, so the model is not ignoring the action; CEM then pulls it back to 0.49, close to the demonstration column. Most residual baselines barely move between Demo and Random, which suggests their mid-horizon error is already dominated by rollout drift. DELTA is the only row that both rises under random actions and recovers under CEM, so the planner is using a model that still grades actions. Taken together, open-loop, replan, and CEM agree on the same split between absolute predictors and residual-plus-state predictors.

5.2 Ablations

Table 3 removes one piece of DELTA at a time on the same test split and the same training budget to completion. The residual head is the large term: without it the model must emit the full token from the latent, open-loop error at h=40 rises from 1.67 to 3.70, and replan at the same horizon stays high (1.15 / 1.33 / 1.76). One-step error also jumps from 0.20 to 0.89, so the failure is not only long-

horizon drift. That matches the latent-only failure mode in Figure 1b: decoding the frozen feature from Z alone starts worse and still rises. No residual is also worse than every main-table residual baseline at h=1 and at h=40, so the ablation is not a mild degradation—it leaves the architecture without the inductive bias that the comparison methods already share. That gap appears under both open-loop and replan, not only on one protocol.

Table 3: Ablations on the same held-out test split. Entries are MSE $\times 10^2$ (lower is bolded). All variants were trained to completion.

Variant	Open-loop MSE ($\times 10^2$)			Replan error at h=40 ($\times 10^2$)		
	h=1	h=10	h=40	r=5	r=10	r=20
Full	**0.20**	**0.45**	**1.67**	**0.52**	**0.67**	**1.23**
No residual	0.89	1.18	3.70	1.15	1.33	1.76
No FiLM	0.21	0.57	1.90	0.65	0.85	1.57
Single stream	0.30	0.70	1.94	0.91	1.30	1.62
Zero action	0.21	0.56	2.22	0.70	0.85	1.62

Without FiLM the action is still added, but no longer scales the tokens (equation 5): one-step error stays near 0.21, while h=40 rises to 1.90 and replan at that horizon rises to 0.65 / 0.85 / 1.57. The amplification therefore matters once the rollout leaves the current frame. Relative to Full, the cost is about 0.12 at h=10 and 0.23 at h=40—smaller than the residual ablation, but enough to erase the mid-horizon margin over MLP-Δ. FiLM is therefore a mid-to-long carry term, not a one-step accuracy knob.

Dropping the dynamics stream leaves one semantic line of K tokens: every open-loop horizon rises (0.30, 0.70, 1.94 at h=1 / 10 / 40), and replan at h=40 rises to 0.91 / 1.30 / 1.62. Single-stream replan at r=10 already sits with TWM and STORM on the main table. Zeroing the action leaves one-step error at 0.21 and raises h=40 to 2.22: a larger long-horizon hit than No FiLM. Under frequent restoration, No residual never recovers (1.15 at r=5), whereas No FiLM and Zero action stay closer to Full and peel away only as r grows. Across the four removals, the residual is necessary for the short horizon; the dynamics stream and the action path matter for the longer carry. None of the ablations undercuts Full, and the ordering of damage—residual, then action/stream, then FiLM—aligns with the Method design.

6 Discussion

Section 5 already assigns each ablation to a horizon band; the design question that remains is what that ranking buys for a frozen backbone. A shared V-JEPA cache makes architecture the only free variable, so a gap at h=1 cannot be blamed on a different encoder, and a CEM recovery cannot be blamed on a different observation stream. That is the sense in which the residual-plus-state split is readable: it is scored where the baselines live, not in a separate latent currency.

That is also why the paper scores feature error as an upstream world-model probe: open-loop, replan, and CEM test drift, recovery, and action ranking under chosen actions, without mixing in a policy or reset distribution. Dreamer-style pool recurrent models and discrete codebook predictors are left out of the main ranking because they do not emit the continuous token field that the metric scores. Within that field the ranking is stable across three protocols; the claim that survives is the short-horizon gap and the replan/CEM order under one cache. Closed-loop task success remains downstream of this feature-space world model.

7 Conclusion

DELTA couples a dual-stream latent with a residual head on frozen V-JEPA 2.1 tokens. On the LIBERO Object split it is lowest or tied under open-loop, replan, and CEM; ablations place short-horizon fidelity on the residual and longer carry on the dynamics stream and FiLM.

AI USE STATEMENT

We used generative AI tools only to edit the manuscript for grammar, wording, and readability. We did not use them to design methods or experiments, implement code, generate or analyze data, derive claims, or interpret results. All AI-assisted text was reviewed by the authors. We take full responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

We provide the model and training objective in Section 3, and the frozen V-JEPA 2.1 feature-cache protocol, LIBERO Object split (seed 0), and open-loop / replan / CEM settings in Section 4. All compared methods share the same cache. Main and ablation numbers are reported in Section 5; ablation definitions appear in Section 3 and Table 3. Anonymous model code and evaluation scripts for the reported protocols are included as supplementary material. Appendix A lists training hyperparameters and extra tables. Trained checkpoints are omitted due to size and will be released upon acceptance, together with training scripts.

REFERENCES

Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in atari. In *Advances in Neural Information Processing Systems*, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/6bdde0373d53d4a501249547084bed43-Abstract.html.

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. 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. doi: 10.1109/CVPR52729.2023.01499. URL https://doi.org/10.1109/CVPR52729.2023.01499.

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

Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. Navigation world models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15791–15801. IEEE, 2025. doi: 10.1109/CVPR52734.2025.01472. URL https://doi.org/10.1109/CVPR52734.2025.01472.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mahmoud Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video, 2024. URL https://arxiv.org/abs/2404.08471.

Jake Bruce, Michael D. Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, Yusuf Aytar, Sarah Maria Elisabeth Bechtle, Feryal Behbahani, Stephanie C. Y. Chan, Nicolas Heess, Lucy Gonzalez, Simon Osindero, Sherjil Ozair, Scott Reed, Jingwei Zhang, Konrad Zolna, Jeff Clune, Nando de Freitas, Satinder Singh, and Tim Rocktäschel. Genie: Generative interactive environments. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 4603–4623. PMLR, 2024. URL https://proceedings.mlr.press/v235/bruce24a.html.

Maxime Burchi and Radu Timofte. Accurate and efficient world modeling with masked latent transformers. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 5894–5912. PMLR, 2025. URL https://proceedings.mlr.press/v267/burchi25a.html.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9630–9640. IEEE, 2021. doi: 10.1109/ICCV48922.2021.00951. URL https://doi.org/10.1109/ICCV48922.2021.00951.

Chang Chen, Yi-Fu Wu, Jaesik Yoon, and Sungjin Ahn. TransDreamer: Reinforcement learning with transformer world models. In *Deep RL Workshop, NeurIPS*, 2021. URL https://arxiv.org/abs/2202.09481.

Fei Deng, Ingook Jang, and Sungjin Ahn. DreamerPro: Reconstruction-free model-based reinforcement learning with prototypical representations. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 4956–4975. PMLR, 2022. URL https://proceedings.mlr.press/v162/deng22a.html.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

David Ha and Jürgen Schmidhuber. World models, 2018. URL https://arxiv.org/abs/1803.10122.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2555–2565. PMLR, 2019. URL https://proceedings.mlr.press/v97/hafner19a.html.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations*, 2020. URL https://arxiv.org/abs/1912.01603.

Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2010.02193.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640(8059):647–653, 2025. doi: 10.1038/s41586-025-08744-2. URL https://doi.org/10.1038/s41586-025-08744-2.

Nicklas Hansen, Hao Su, and Xiaolong Wang. TD-MPC2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/cf73d57b6dcda32b293df7c2d5341f49-Abstract-Conference.html.

Nicklas A. Hansen, Hao Su, and Xiaolong Wang. Temporal difference learning for model predictive control. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 8387–8406. PMLR, 2022. URL https://proceedings.mlr.press/v162/hansen22a.html.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15979–15988. IEEE, 2022. doi: 10.1109/CVPR52688.2022.01553. URL https://doi.org/10.1109/CVPR52688.2022.01553.

Michael Janner, Qiyang Li, and Sergey Levine. Offline reinforcement learning as one big sequence modeling problem. In *Advances in Neural Information Processing Systems*, 2021. URL https://papers.neurips.cc/paper_files/paper/2021/file/099fe6b0b444c23836c4a5d07346082b-Paper.pdf.

Michael Janner, Yilun Du, Joshua Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 9902–9915. PMLR, 2022. URL https://proceedings.mlr.press/v162/janner22a.html.

Thomas Kipf, Elise van der Pol, and Max Welling. Contrastive learning of structured world models. In *International Conference on Learning Representations*, 2020. URL https://arxiv.org/abs/1911.12247.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 44776–44791, 2023. doi: 10.52202/075280-1939. URL https://doi.org/10.52202/075280-1939.

Vincent Micheli, Eloi Alonso, and François Fleuret. Transformers are sample-efficient world models. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=vhFu1Acb0xb.

Vincent Micheli, Eloi Alonso, and François Fleuret. Efficient world models with context-aware tokenization. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 35623–35638. PMLR, 2024. URL https://proceedings.mlr.press/v235/micheli24a.html.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=a68SUt6zFt. arXiv:2304.07193.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), 2018. doi: 10.1609/aaai.v32i1.11671. URL https://doi.org/10.1609/aaai.v32i1.11671.

Jan Robine, Marc Höftmann, Tobias Uelwer, and Stefan Harmeling. Transformer-based world models are happy with 100k interactions. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=gb6ocYuVhk1.

Reuven Rubinstein. The cross-entropy method for combinatorial and continuous optimization. *Methodology and Computing in Applied Probability*, 1(2):127–190, 1999. doi: 10.1023/A:1010091220143. URL https://doi.org/10.1023/A:1010091220143.

Younggyo Seo, Danijar Hafner, Hao Liu, Fangchen Liu, Stephen James, Kimin Lee, and Pieter Abbeel. Masked world models for visual control. In *Proceedings of The 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pp. 1332–1344. PMLR, 2023. URL https://proceedings.mlr.press/v205/seo23a.html.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. VideoMAE: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *Advances in Neural Information Processing Systems*, volume 35, pp. 10078–10093, 2022. doi: 10.52202/068431-0732. URL https://doi.org/10.52202/068431-0732.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017. URL https://papers.nips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Philipp Wu, Alejandro Escontrela, Danijar Hafner, Pieter Abbeel, and Ken Goldberg. DayDreamer: World models for physical robot learning. In *Proceedings of The 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pp. 2226–2240. PMLR, 2023. URL https://proceedings.mlr.press/v205/wu23c.html.

Sherry Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Leslie Kaelbling, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/c4d66eae503694424123b93ac0fbaf17-Paper-Conference.pdf.

Weipu Zhang, Gang Wang, Jian Sun, Yetian Yuan, and Gao Huang. STORM: Efficient stochastic transformer based world models for reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 27147–27166, 2023. doi: 10.52202/075280-1182. URL https://doi.org/10.52202/075280-1182.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. DINO-WM: World models on pre-trained visual features enable zero-shot planning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 79115–79135. PMLR, 2025. URL https://proceedings.mlr.press/v267/zhou25t.html.

Yuke Zhu, Josiah Wong, Ajay Mandlekar, Roberto Martín-Martín, Abhishek Joshi, Kevin Lin, Abhiram Maddukuri, Soroush Nasiriany, and Yifeng Zhu. robosuite: A modular simulation framework and benchmark for robot learning, 2020. URL https://arxiv.org/abs/2009.12293. Technical report. The live arXiv BibTeX export currently prints 2025 because the file was revised; this entry uses the 2020 release year.

A Additional experimental details

This appendix records settings and tables that support the main text. It does not change the ranking in Section 5. All feature errors below use the same units as the main tables (MSE $\times 10^2$).

A.1 Training hyperparameters

DELTA is trained with Adam at learning rate 10^{-4}, batch size 16, window length $H{=}16$, and rollout length $k{=}8$ in equation 11. The architecture matches Section 3: $d{=}256$, $K{=}32$, four heads, $L{=}2$ transition blocks, and a residual feature head. Loss weights are $\lambda_{\text{roll}}{=}1$ and $\lambda_{\text{reg}}{=}1$, with $\tau{=}0.015$ in equation 12. The reported DELTA checkpoint is taken from a run budgeted at 5000 optimizer steps and evaluated at step 2600; ablations use the same optimizer and architecture flags with a 2500-step budget and are trained to completion. Baselines share the same frozen feature cache and Adam / 10^{-4} / batch-16 defaults, and are likewise trained to completion under comparable step budgets. Checkpoints for tables are not selected on the reported evaluation set.

A.2 Data split (coarse)

LIBERO Object demonstrations are split task-wise with seed 0 into 364 training, 70 validation-monitoring, and 20 held-out test episodes, as in Section 4.1. Tables report means on that held-out test split. We do not list episode indices here; the split script is in the code supplement.

A.3 Excluded predictors

Table 4 records two predictors that we do not put in the main ranking.

IRIS-VQ is a discrete codebook model trained on the same cache and scored with continuous semantic token MSE. Its open-loop error stays near 70 under the $\times 10^2$ units of the main tables—two orders of magnitude above every continuous residual row. That gap is why discrete codebook models are left out of Table 1: the metric is a continuous feature field.

A Dreamer-style pool RSSM predicts only a single pooled vector. To force it onto the same K-token open-loop score we broadcast that vector to all K tokens. The resulting error sits next to a spatial floor obtained by broadcasting the mean target token (Table 4): the broadcast has almost no spatial structure, so the comparison mainly shows "token head vs. no token head," not a fair dynamics race.

Pool-only recurrent models are therefore left out of the main table for the same reason stated in Section 4.2.

Table 4: Predictors excluded from the main ranking (MSE $\times 10^2$). Dreamer rows use an unfair broadcast of a pool prediction onto $K{=}32$ tokens; the floor broadcasts the mean target token. DELTA from Table 1 is repeated for scale.

Method	$h{=}1$	$h{=}5$	$h{=}10$	$h{=}20$	$h{=}40$
DELTA (Table 1)	0.20	0.35	0.45	0.92	1.67
Dreamer (broadcast pool$\rightarrow K$)	7.52	7.47	7.64	7.51	7.86
Spatial floor (broadcast mean)	7.36	7.37	7.53	7.42	7.76
IRIS-VQ	70.72	70.69	70.64	70.11	70.39

A.4 Ablation open-loop horizons

Table 5 expands Table 3 to every reported open-loop horizon. The residual removal remains the dominant term at every h; FiLM, single stream, and zero action mainly hurt mid-to-long horizons.

Table 5: Ablation open-loop semantic token MSE ($\times 10^2$) on the same held-out test split.

Variant	$h{=}1$	$h{=}5$	$h{=}10$	$h{=}20$	$h{=}40$
Full	**0.20**	**0.35**	**0.45**	**0.92**	**1.67**
No residual	0.89	1.03	1.18	1.66	3.70
No FiLM	0.21	0.38	0.57	1.07	1.90
Single stream	0.30	0.47	0.70	1.26	1.94
Zero action	0.21	0.38	0.56	1.11	2.22