LEARNING GRAVITY FROM VIDEO: STRUCTURED LATENT STATES FOR PREDICTION AND INTERVENTION

Anonymous authors
Paper under double-blind review

ABSTRACT

A physical system evolves according to its current state and governing dynamics. This suggests organizing video world models around visually inferred states and reusable evolution rules. Yet predictive learning alone does not ensure that learned representations support sustained physical evolution and meaningful intervention. We introduce GravJEPA, a self-supervised world model built on LeWorldModel that separates visual state estimation from dynamical evolution. The model infers structured latent components from video histories and propagates them with a time-homogeneous Koopman-form transition. This formulation supports direct queries at future times after observations stop. To align the inferred states with this dynamical structure, we introduce regularization objectives that encourage translation equivariance and photometric invariance in position-associated representations. The visual states are learned without ground-truth physical-state or gravity supervision. On controlled gravity videos, GravJEPA achieves lower trajectory prediction errors than an autoregressive baseline trained with the same equivariance and invariance regularization. It also outperforms this baseline in long-horizon prediction and out-of-distribution tests with unseen velocity and gravity magnitudes. Beyond prediction, matched interventions on inferred velocity- and gravity-associated components produce corresponding changes in future motion through the same transition. Together, these results establish a shared latent-state interface for forecasting and intervention, taking a step toward physics-engine-like world models learned from video.

1 INTRODUCTION

Our world operates under universal physical laws that govern diverse dynamic events. Under a uniform gravitational field, varying initial positions and velocities yield distinct trajectories, whereas shifting the observation window or visual appearance alters sensory input without changing underlying dynamics. Effective world models should thus infer visual states that evolve under a shared dynamical law across initial conditions, temporal anchors, and visual variations (Champion et al., 2019; Lusch et al., 2018). Such models enable accurate long-horizon predictions after observations cease (Hafner et al., 2019; Wu et al., 2022), which is crucial for downstream physical reasoning and interaction (Motamed et al., 2026; Bansal et al., 2025).

Recent evaluations demonstrate that video generation models predicting well under familiar conditions often fail to generalize outside their training distributions (Kang et al., 2025). An alternative to pixel-level generation is to learn predictive representations through joint-embedding predictive architectures (JEPA) (Assran et al., 2023; Bardes et al., 2024; Assran et al., 2025). Building on this, LeWorldModel (LeWM) (Maes et al., 2026) pairs next-embedding prediction with distribution regularization for pixel-to-latent learning. While theoretical results guarantee identifiability under specific latent assumptions (Klindt et al., 2026), preventing representation collapse and achieving low predictive error do not ensure that latent states retain necessary motion dynamics (Tan et al., 2026; Zhang et al., 2026). A globally valid representation may remain overly sensitive to visual appearance while failing to distinguish physical configurations or action consequences (Zeng et al., 2026).

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.

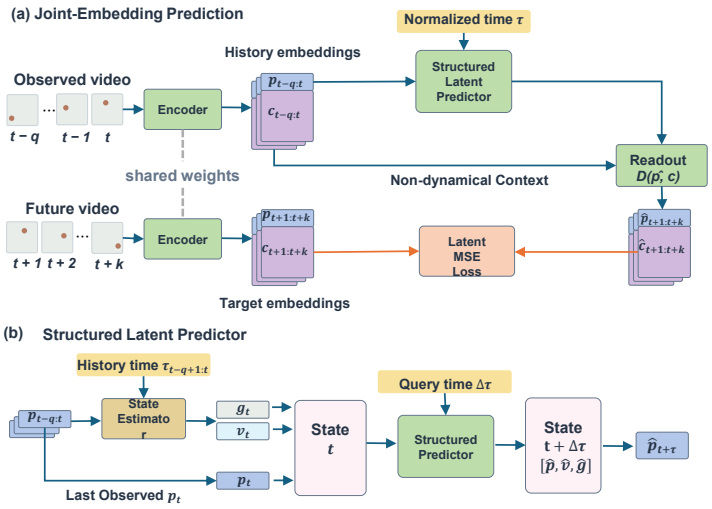


Figure 1: Model overview. **(a)** A shared frame encoder produces history and target embeddings. The structured latent predictor evolves the dynamical coordinates to a requested future time, while non-dynamical context is passed to the readout for future-feature prediction. **(b)** The inference network estimates velocity- and gravity-associated components from the encoded history and combines them with the last observed position-associated coordinate. A fixed constant-acceleration transition then propagates the resulting state to the query time.

Even a prescribed evolution rule only constrains how latent coordinates evolve; it does not determine how visual observations should be mapped into them. Consequently, learning a compatible state from video remains nontrivial. Inferring physically meaningful coordinates without explicit supervision is further complicated by non-identifiability (Locatello et al., 2019; Wang et al., 2026). Without ground-truth physical labels, the functional roles of position, velocity, and acceleration must emerge through architectural priors and training objectives. Effective information flow must map finite visual histories into evolvable latent states, as seen in models that isolate latent kinematics before propagating dynamics (Yıldız et al., 2019). Furthermore, transformation-based representation learning helps determine which variations a model preserves or suppresses (Garrido et al., 2023; 2024), motivating objectives that disentangle physical configurations from visual appearance. Beyond decodability, inferred states must be functionally active: after observations end, state propagation should yield accurate trajectory rollouts, and latent interventions should produce corresponding physical changes (Janny et al., 2022).

To address these challenges, we introduce **GravJEPA**, a structured joint-embedding video world model that decouples visual state inference from dynamics propagation. Given a short observation history, GravJEPA infers position-, velocity-, and gravity-associated latent components. Drawing on Koopman-based representation learning (Lusch et al., 2018; Otto & Rowley, 2019), a shared, predefined constant-acceleration transition evolves the inferred state to arbitrary future query times, while a context-conditioned readout maps the evolved coordinates to future representations. The model is trained without ground-truth physical-state supervision.

We further introduce *equivariance–invariance regularization* (EIR) to shape the visual coordinates used for state inference. EIR encourages translation equivariance and photometric invariance: spatial translations should induce corresponding displacements in the position-associated coordinates, whereas controlled photometric transformations should leave these coordinates unchanged (Locatello et al., 2020). Combined with history fitting and predictive losses (Jaques et al., 2019; Guen & Thome, 2020), as well as distribution regularization, EIR encourages the encoder and state estimator to form coordinates that are compatible with the prescribed transition.

We evaluate both the information content and functional utility of the learned states. Linear probes assess physical-variable decodability, while open-loop forecasts assess long-horizon prediction. To test functional causality, we perform targeted edits on motion-related latent components and observe whether they induce intended physical changes (Li et al., 2022). Complementary appearance probes examine how visual information is distributed across the dynamical coordinates and memory (Denton & Birodkar, 2017; Li & Mandt, 2018).

Our contributions are summarized as follows:

Structured latent states for action-free world modeling: We introduce **GravJEPA**, an action-free joint-embedding video world model that infers position-, velocity-, and gravity-associated latent components from visual histories and evolves them with a shared Koopman-form transition.

EIR for mitigating shortcut learning: We introduce Equivariance–Invariance Regularization (EIR) to mitigate shortcut learning in our JEPA-based world model. EIR encourages position-associated coordinates to be equivariant to spatial translations and invariant to photometric transformations, shaping the visual state used by the structured predictor without physical-state supervision.

Predictive and interventional state validation: We evaluate open-loop forecasting across unseen physical conditions, long-horizon and out-of-view settings, and varied observation and query times. We further examine where physical-state and appearance information is encoded, how motion states are organized, and whether targeted latent-state interventions induce the expected changes in predicted motion.

2 Related Work

Predictive representations and state formation. Joint-embedding methods learn predictive image and video representations rather than reconstruct pixels (Assran et al., 2023; Bardes et al., 2024; Assran et al., 2025). LeWM brings this approach to end-to-end world modeling through next-embedding prediction and Gaussian regularization (Maes et al., 2026). However, the physical information retained depends on both observations and prediction targets (Tan et al., 2026). Semigroup-JEPA uses recursive-rollout training and explicitly supplied gravity for action-free physical prediction (Liu et al., 2026a). We instead infer dynamical states from visual histories without ground-truth gravity inputs during training or inference.

State estimation and structured latent dynamics. Recurrent world models combine observation-conditioned state inference with latent imagination (Hafner et al., 2019; 2020; Li et al., 2026b). Structured alternatives include continuous-time and linear latent dynamics (Chen et al., 2018; Lusch et al., 2018), as well as reconstruction-based models with second-order states or time-invariant dynamics and appearance modulators (Yıldız et al., 2019; Auzina et al., 2023). For video prediction, ODEWorld integrates a time- and context-conditioned field over frozen visual features (Liu et al., 2026b), whereas LDR combines finite-difference initialization with fixed integration and learned higher-order residuals under image and latent losses (Li et al., 2026a). Our focus is history-based state estimation within a task-specialized JEPA, with editable velocity- and gravity-associated components propagated by a shared time-homogeneous transition.

Physical prediction and intervention. Physical-video research studies generalization in simulation (Kang et al., 2025), counterfactual prediction from intervention-paired videos without ground-truth position supervision (Janny et al., 2022), and parameter identifiability for a specified second-order linear ODE family under state-consistency and trajectory-coverage assumptions (Wang et al., 2026). PhyLatent and PSG-JEPA add training-time grounding through simulator states and robot proprioceptive signals, respectively (Zeng et al., 2026; Yan et al., 2026). We evaluate open-loop physical prediction and interventions on inferred velocity- and gravity-associated components, measuring intended trajectory changes and unintended effects on quantities that should remain unchanged.

3 Method

We introduce **GravJEPA**, a joint-embedding video world model that predicts future representations by evolving states inferred from visual histories. Building on LeWM (Maes et al., 2026), GravJEPA replaces autoregressive feature prediction with a structured latent predictor and a context-conditioned readout. Equivariance–invariance regularization (EIR) guides the visual coordinates used for state inference by distinguishing spatial displacement from photometric variation.

We consider gravity-driven motion without contact or changes in the gravitational regime during the modeled interval. Training uses video frames, timestamps, and known image transformations, without ground-truth position, velocity, or gravity labels. At inference, the model receives an observation history and requested future times.

3.1 JOINT-EMBEDDING PREDICTION

For observations $\{(x_i, t_i)\}_{i=1}^{K}$ ending at $t_\star = t_K$, a shared encoder maps each frame to

$$z_i = E_\theta(x_i) = [p_i, c_i], \qquad p_i \in \mathbb{R}^d, \quad c_i \in \mathbb{R}^{D-d}. \tag{1}$$

The first $d = D/4$ coordinates, p_i, provide the input to the structured latent predictor. The remaining coordinates c_i supply complementary visual information. We denote the non-dynamical context passed from the observed embeddings to the readout by c. For a requested future time u, the model predicts

$$\hat{z}(u) = D_\omega(\hat{p}(u), c)\,, \tag{2}$$

where $\hat{p}(u)$ is obtained by evolving the inferred state. The context c is held fixed during prediction rather than propagated through the dynamical transition. This separates explicit state evolution from the contextual information used to predict the complete embedding. The same encoder produces the future target representations during training.

3.2 STRUCTURED LATENT PREDICTION

Inferring the state. The last observed coordinate $p_\star = p_K$ anchors the current position, while the preceding coordinate changes determine the inferred motion. We express time relative to the observation cutoff as $\tau(t) = (t - t_\star)/\mu$, where $\mu = (t_K - t_1)/(K-1)$. With $\rho = \mathrm{vec}[p_1 - p_\star, \ldots, p_K - p_\star]$, the state estimator gives

$$(v_\star, g_\star) = H_\psi(\rho, \eta(\tau_{1:K}))\,, \qquad s_\star = [p_\star^\top, v_\star^\top, g_\star^\top]^\top, \tag{3}$$

where $\tau_i = \tau(t_i)$ and η stacks the fixed time encodings. The components $v_\star, g_\star \in \mathbb{R}^d$ represent latent velocity and acceleration on this normalized time scale.

Evolving the state. GravJEPA uses a fixed constant-acceleration flow in Koopman form:

$$K_{\Delta\tau} = \exp(A\Delta\tau), \qquad A = \begin{bmatrix} 0 & I_d & 0 \\ 0 & 0 & I_d \\ 0 & 0 & 0 \end{bmatrix}. \tag{4}$$

The same generator A is shared across videos, so differences in motion are represented by the inferred state rather than different transition parameters. Its position block yields

$$\hat{p}(u) = \Pi_p K_{\tau(u)} s_\star = p_\star + \tau(u) v_\star + \tfrac{1}{2}\tau(u)^2 g_\star, \qquad \Pi_p = [I_d\ 0\ 0]. \tag{5}$$

Future states can therefore be queried directly without re-encoding predicted features. For a fixed state and time normalization, $K_{a+b} = K_b K_a$ makes direct and successive propagation consistent. The transition prescribes the evolution form; the encoder and estimator learn coordinates and states to be used by that transition.

Editing the state. Prediction and intervention use the same state interface. Replacing $v_\star$ or $g_\star$ while keeping $p_\star$, c, the transition, and the readout fixed gives

$$s_\star^{\mathrm{edit}} = [p_\star^\top, {v'_\star}^\top, {g'_\star}^\top]^\top, \qquad \hat{z}^{\mathrm{edit}}(u) = D_\omega\big(\Pi_p K_{\tau(u)} s_\star^{\mathrm{edit}}, c\big)\,. \tag{6}$$

The corresponding coordinate change is $\delta p(u) = \tau(u)\delta v + \frac{1}{2}\tau(u)^2 \delta g$. These are edits in learned coordinates; their correspondence to physical interventions is evaluated separately.

3.3 EQUIVARIANCE–INVARIANCE REGULARIZATION

The transition specifies how coordinates evolve, but not how they should respond to visual changes. EIR imposes two complementary requirements on the position-associated coordinates:

$$\begin{aligned} &p(T_\delta x) \approx p(x) + Q\delta \quad \text{(translation equivariance)}, \\ &p(C_\xi x) \approx p(x) \qquad\quad \text{(photometric invariance)}. \end{aligned} \tag{7}$$

Here $p(x)$ denotes the first d coordinates of $E_\theta(x)$. The whole-frame translation T_δ moves the image, including its background, by a known normalized displacement δ. The photometric transform C_ξ changes brightness, contrast, saturation, and channel gains without changing pixel locations. Each transform uses the same parameters across the frames to which it is applied within a clip. Considering applicability to natural videos, we design EIR to use only generic frame-level transformations and their known parameters, requiring neither simulator states nor object-level annotations.

A learned map $Q \in \mathbb{R}^{d\times 2}$, constrained by $Q^\top Q = I_2$, maps image displacements to latent offsets without collapsing the displacement map to zero.

We penalize deviations from these relations using a displacement-normalized translation loss $\mathcal{L}_{\text{eq}}$ and a photometric consistency loss $\mathcal{L}_{\text{photo}}$:

$$\mathcal{L}_{\text{EIR}} = \lambda_e \mathcal{L}_{\text{eq}} + \lambda_c \mathcal{L}_{\text{photo}}. \tag{8}$$

Both losses act only on p, leaving the complementary coordinates unconstrained by EIR. Exact loss reductions, frame selection, and displacement normalization are given in Appendix A.

The translation constraint complements the relative-history estimator: when $p_i^S = p_i^O + Q\delta$ holds throughout a history, $p_i^S - p_K^S = p_i^O - p_K^O$. At unchanged timestamps, the estimator then receives the same relative history and infers the same $v_\star, g_\star$, while $p_\star$ shifts. Photometric invariance similarly encourages the motion-estimation input to remain unchanged under appearance changes.

3.4 Training Objective

Prediction and history fitting. The inferred state is trained to explain the observed coordinate history and predict future features. Let Z^+ contain the encoded future targets and $\hat{Z}^+_{\text{core}}$ the future features used in the dynamical core's prediction objective. Let P^- contain the observed p_i, and let $\hat{P}^-$ contain their reconstructions $\Pi_p K_{\tau_i} s_\star$. We use

$$\mathcal{L}_{\text{pred}} = \text{MSE}(\hat{Z}^+_{\text{core}}, Z^+), \qquad \mathcal{L}_{\text{hist}} = \text{MSE}(\hat{P}^-, P^-), \tag{9}$$

where MSE averages over batch, time, and feature dimensions. The history term requires a single inferred state to account for the observed motion rather than fitting future targets alone.

Joint training. The dynamical-core objective combines these losses with EIR and full-feature distribution regularization:

$$\mathcal{L}_{\text{core}} = \mathcal{L}_{\text{pred}} + \lambda_h \mathcal{L}_{\text{hist}} + \mathcal{L}_{\text{EIR}} + \lambda_r \mathcal{L}_{\text{reg}}, \tag{10}$$

where $\mathcal{L}_{\text{reg}}$ is VISReg (Wu et al., 2026) applied to the full original-video embeddings. The auxiliary memory objective trains the context-conditioned prediction branch without updating the dynamical core. Both branches are trained from the first update. The shared encoder also produces the future targets, and the core objective propagates gradients through these targets without an EMA teacher. Future frames are used only as training targets and in training losses, not as inputs to state inference. Appendix A provides the implementation and loss weights.

4 Experiments

We test whether states inferred from a short video history support accurate prediction after observations stop. We organize the experiments around open-loop prediction and state readout, physical and visual distribution shifts, temporal extrapolation and future-state queries, and finally motion-state organization and latent-state interventions.

Datasets. We construct a single-object video benchmark with 64×64 RGB observations. Each trajectory follows two-dimensional constant-acceleration motion under a gravity vector that varies across trajectories but remains fixed within each clip. Object appearance varies across three shapes, eight colors, and three sizes on a uniform gray background, and remains fixed within each clip. The training set contains 11,520 physical trajectories, with four clips per trajectory, yielding 46,080 clips. The validation and test sets contain 576 trajectories each. No complete position–velocity–gravity tuple is shared across splits. During training, we use seven observed frames and six future target frames, with a mixture of uniform and irregular observation intervals. For evaluation, we define a reference horizon $H_{\text{ref}} = 0.33\,\text{s}$ using six future queries spaced 55 ms apart after the last observation. We use this evaluation protocol unless otherwise stated. Appendix B provides the physical ranges, rendering settings, and temporal sampling details.

Models and training. Our LeWM-based autoregressive Transformer baseline, **AR**, uses six-step free-running training and applies VISReg (Wu et al., 2026) instead of SIGReg (Balestriero & LeCun, 2025). **AR + EIR** additionally applies our equivariance–invariance regularization (EIR) to the first quarter of the coordinates in each frame embedding, encouraging translation equivariance and photometric invariance in these coordinates. **GravJEPA** uses the same EIR objectives and combines a structured state estimator. The auxiliary memory branch and dynamical core are trained from the first update, but gradients from the memory objective do not update the core. All models use the same ViT encoder architecture and are trained from scratch for 40,000 updates with a batch size of 256. Physical states are not used as training targets.

(a) Prediction accuracy and model size

Model	ADE ↓	FDE ↓	Params.
AR	11.33 ± 0.02	18.59 ± 0.02	65,456
AR + EIR	2.45 ± 0.62	4.36 ± 0.89	65,464
GravJEPA	$\mathbf{1.81 \pm 0.14}$	$\mathbf{3.58 \pm 0.44}$	50,184

(b) Latent-state probing (R^2)

Model	Frame-pos.	p ind.	v ind.	g ind.
AR	-0.003	–	–	–
AR + EIR	0.997	–	–	–
GravJEPA	0.997	0.995	0.991	0.524

Table 1: **Open-loop prediction and latent-state probing. (a)** Prediction under irregular-ID timing at the fixed 40,000-update checkpoint; ADE and FDE are mean ± standard deviation across three training seeds and are measured in input pixels. Parameter counts include all registered components. **(b)** Linear-probe R^2. Frame-pos. uses the full frame representation, while p, v, and g use the respective structured components. A dash denotes an unavailable result, not a zero score.

Evaluation Protocol. Unless otherwise stated, each model receives seven observed frames and predicts without additional images. We fit standardized linear ridge readouts on the validation set and freeze them across test conditions. The main position readout uses the full frame representation. Average displacement error (ADE) is the mean Euclidean position error across evaluated trajectories and future query times. Final displacement error (FDE) is the mean Euclidean error at the final query time. The default evaluation uses six queries over H_{ref}. Errors are reported in input pixels, with 16 pixels per world-coordinate unit. Table 1(a) evaluates the models from each of three independent training seeds and reports the mean and standard deviation across seeds. Appendix C gives the readout and uncertainty protocols.

4.1 Open-Loop Prediction from Visual Histories

Table 1(a) compares the three models under irregular observation timing drawn from the training distribution. Averaged across three training seeds, GravJEPA reduces ADE from 2.45 to 1.81 pixels and FDE from 4.36 to 3.58 pixels relative to the AR + EIR baseline, reductions of 26.12% and 17.89%, respectively. Both models have position-readout R^2 of approximately 0.997, yet their future prediction errors differ. High position decodability therefore does not alone account for the forecasting performance in this comparison.

Without EIR, AR has poor linear position readout and much larger prediction errors. Its feature variance and effective rank remain nonzero, so this result should not be interpreted as complete representation collapse. The stronger AR + EIR control shows that transformation-specific learning matters, while the remaining gap favors GravJEPA. This comparison does not isolate the contribution of the transition, history fitting, or memory individually.

Latent-state readout. Independent linear probes evaluate whether physical position, velocity, and gravity can be decoded from the inferred p, v, and g components. Position and velocity are highly decodable, with R^2 values of 0.995 and 0.991, while gravity is less strongly decoded at 0.524 (Table 1(b)). Functional use is tested separately in Section 4.5.

4.2 Generalization to Unseen Velocity and Gravity Magnitudes

We evaluate low- and high-magnitude velocity and gravity shifts outside the training support. Directions, cutoff positions, appearances, timestamps, and sampling quantiles are paired across conditions, while all models and the validation-fitted position readout remain frozen. Appendix D provides the exact ranges, paired-set construction, and filtering protocol.

Table 2(a) summarizes performance under unseen velocity and gravity magnitudes. GravJEPA achieves the lowest ADE and FDE in both cases. Under the joint high-velocity and high-gravity shift, it further reduces FDE from 8.98 to 5.31 pixels. Full condition-wise results appear in Appendix D.

4.3 Long-Horizon and Out-of-View Prediction

We extend prediction to $2H_{\text{ref}}$ and $4H_{\text{ref}}$ at the same 55 ms query interval, without additional observations. The models and readouts remain fixed. All 576 trajectories are retained, including those leaving the camera view.

Long-horizon accuracy. GravJEPA achieves lower aggregate ADE and FDE at all evaluated horizons (Table 2(b)). Relative to the **AR + EIR** baseline, it reduces FDE from 36.91 to 17.92 pixels at $2H_{\text{ref}}$ and from 119.31 to 76.44 pixels at $4H_{\text{ref}}$. Thus, its ability to predict beyond the image boundary is accompanied by lower aggregate long-horizon error.

(a) Magnitude OOD

	Velocity OOD		Gravity OOD	
Model	ADE	FDE	ADE	FDE
AR	11.93	20.03	11.92	20.13
AR + EIR	4.09	7.18	3.41	5.78
GravJEPA	**2.00**	**4.31**	**2.05**	**4.41**

(b) Long-horizon prediction

	$2H_{\text{ref}}$		$4H_{\text{ref}}$	
Model	ADE	FDE	ADE	FDE
AR	22.07	41.97	49.22	112.87
AR + EIR	11.60	36.91	48.46	119.31
GravJEPA	**6.67**	**17.92**	**26.43**	**76.44**

Table 2: **Physical OOD and long-horizon prediction.** **(a)** Macro-average over low/high velocity- and gravity-magnitude OOD at H_{ref}. **(b)** Prediction at longer horizons using 55 ms queries; H_{ref} results appear in Table 1(a). ADE and FDE are in input pixels (↓). Full OOD settings and condition-wise results appear in Appendix D.

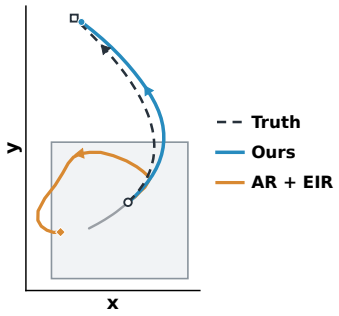
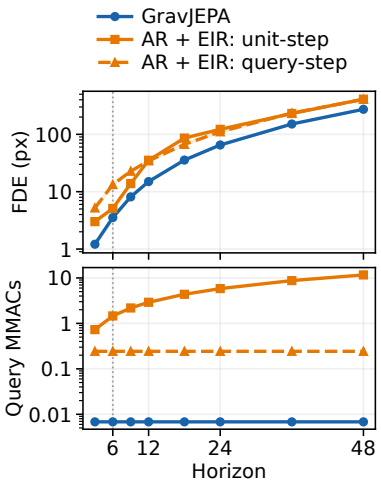


Figure 2: **Out-of-view trajectory prediction.** Shaded boxes mark the camera view; gray curves show the observed histories. GravJEPA (blue) and AR + EIR (orange) are compared against ground truth (dashed).

(a) Temporal queries



(b) Functional differentiation

Linear / MLP (%)

Component	Color	Shape
GravJEPA		
p	35.6/34.1	51.4/44.9
rest	35.1/33.6	48.6/57.2
full	37.5/33.6	47.6/55.6
memory	88.4/88.8	52.8/63.4
AR+EIR		
p	16.8/17.4	54.9/63.3
rest	16.4/24.5	62.2/62.7
full	16.1/25.7	62.8/62.8
AR without EIR		
full	100.0/100.0	56.6/77.2

Table 3: **Temporal queries and latent functional differentiation.** **(a)** Endpoint FDE (top) and cached-query matrix MACs (bottom), both on log scales, from seven observed frames. The dotted line marks the H6 training horizon (0.055 s/step). **(b)** Color/shape classification accuracy (%), reported as linear/MLP. Chance levels are 14.3%/33.3%. MLP results average three probe initializations on one frozen checkpoint per model. AR+EIR's first four coordinates receive EIR but are not an explicit $p/v/g$ factorization; full denotes the complete 16-dimensional embedding.

Out-of-view prediction. GravJEPA can extrapolate trajectories beyond the camera view, whereas the AR + EIR baseline tends to keep predictions within or return them to the visible region (Figure 2).

Query efficiency. The same state inferred at the observation cutoff can answer queries at multiple future times directly. Across the evaluated query times, GravJEPA yields lower endpoint error than query-step AR + EIR while using approximately 35.6× fewer counted matrix MACs per cached query (Table 3(a)). This factor measures only cached-query matrix MACs, not end-to-end wall-clock speedup; in particular, it excludes history encoding and state inference. Query-step AR + EIR also has horizon-independent query cost, whereas unit-step AR requires increasing computation as the requested horizon grows.

4.4 Functional Differentiation across Latent Components

We probe color and shape information in single-frame components while holding the physical context fixed; Appendix F.1 provides the full protocol. Table 3(b) shows substantially stronger color

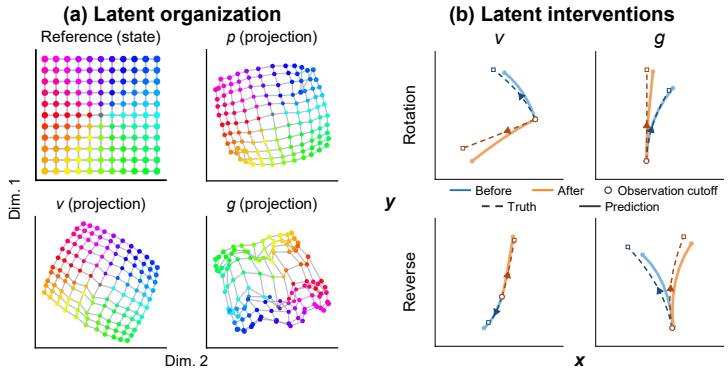


Figure 3: **Latent-state organization and interventions.** **(a)** Reference grid and t-SNE visualizations of the position-, velocity-, and gravity-associated components p, v, and g. Colors indicate grid locations, and edges connect neighboring physical conditions. **(b)** Editing v or g (columns) changes the model's predicted future motion. The edits correspond to a 90° rotation or reversal of the respective physical vector (rows), while the transition and readout remain unchanged. Blue and orange indicate the original and counterfactual futures, respectively; dashed and solid lines show ground-truth and predicted trajectories. Open circles mark the observation cutoff.

decodability in GravJEPA's memory than in p, whereas shape allocation is less distinct. The p component remains strongly predictive of position ($R^2 = 0.967/0.971$ on the color/shape sets) but is still appearance-readable.

AR+EIR has no explicit $p/v/g$ factorization, yet its full embedding supports both position readout ($R^2 = 0.996/0.995$) and appearance classification. Thus, the results indicate partial functional specialization rather than complete disentanglement or appearance invariance.

While the AR representation retains nonzero variance and effective rank, its position readout is near zero ($R^2 = -0.003$; Table 1(b)), while color is perfectly decodable (Table 3(b)). Thus, without EIR, AR exhibits poor linear spatial decodability, preferentially retaining stable color information rather than position.

4.5 Motion-State Organization and Functional Interventions

We assess the learned motion states from two complementary perspectives: controlled physical sweeps test how the inferred components are organized, while component replacement tests whether they functionally control predicted motion.

Controlled state grids. Following the grid-based visualization in LeWM (Maes et al., 2026), we visualize the position-associated component p and extend the approach to v and g using controlled video histories. We generate three sets of 64×64 videos, varying physical position, velocity, or gravity one at a time. In each set, the chosen variable takes values on an 11×11 grid. The other two variables have the same values across videos at the observation cutoff, and appearance is kept unchanged. We then visualize the corresponding latent component p, v, or g to examine how it responds to these physical changes. Figure 3(a) shows t-SNE projections of the inferred p, v, and g components.

Latent-state interventions. We test whether replacing an inferred velocity or gravity component produces the corresponding change in predicted motion. For each source trajectory, we render matched donor histories by zeroing, reversing, rotating by 90°, or scaling by 1.5 either physical velocity or gravity. The current position, the other physical vector, appearance, and observation times remain unchanged. We then replace only the corresponding inferred latent component, keeping the source position latent, the other dynamical component, and appearance memory fixed. Thus, these operations define physical changes used to construct donor histories, rather than direct transformations of latent coordinates. The model and the position probe fitted on the calibration split remain frozen throughout evaluation.

We measure counterfactual endpoint error and the response-error ratio

$$\mathcal{E}_{\text{resp}} = \frac{1}{N}\sum_{i=1}^{N}\frac{\|\Delta\hat{\mathbf{q}}_i - \Delta\mathbf{q}_i\|_2^2}{\|\Delta\mathbf{q}_i\|_2^2}, \tag{11}$$

where $\Delta\hat{\mathbf{q}}_i$ and $\Delta\mathbf{q}_i$ denote predicted and true edited-minus-unedited endpoint displacements. An exact response gives zero, while ignoring the edit gives one. We compute this metric separately for each transformation and average equally across the four transformations.

Matched single-component replacement produces motion changes closer to the intended interventions than shuffled-donor controls, with more accurate responses for velocity than gravity. Figure 3(b) illustrates rotation and reversal results. Full quantitative results are provided in Appendix F.3.

These results support functional control through the inferred velocity and gravity components under matched simulator interventions. They do not establish fully disentangled physical coordinates or reliability across training seeds.

4.6 Ablations on state formation and prediction

We ablate history fitting, translation equivariance, and photometric consistency to assess their roles in prediction and state formation. Table 4 reports the mean results. These objective ablations use the smaller-scale configuration described in Appendix G. Panel (a) compares prediction and state readout, while Panel (b) examines the effect of photometric consistency under controlled recoloring.

(a) Prediction and state readout

Variant	ADE ↓	FDE ↓	p ↑	v ↑	g ↑
Full	**4.78**	**9.15**	.98	.95	.37
w/o $\mathcal{L}_{\text{hist}}$	7.91	13.09	.99	.95	.47
w/o $\mathcal{L}_{\text{eq}}$	11.87	19.67	.00	−.01	−.01
w/o $\mathcal{L}_{\text{photo}}$	5.89	10.64	.98	.95	.48

(b) Controlled recoloring

Variant	Current drift ↓	Future drift ↓	FDE ↓
Full	.24	1.61	13.10
w/o $\mathcal{L}_{\text{photo}}$	.44	1.24	11.35

Table 4: Training-objective ablations at H_{ref}. **(a)** ADE and FDE are measured in input pixels; the $p/v/g$ columns report R^2 from separate frozen linear probes. **(b)** The same trajectory is rendered in eight colors. Current and future drift measure pairwise changes in decoded position across colors at the observation cutoff and H_{ref}, respectively. FDE measures error against ground truth; all three are in input pixels.

Removing translation equivariance causes the position, velocity, and gravity probe scores to fall to approximately zero and increases ADE from 4.78 to 11.87 pixels. Thus, future prediction and distribution regularization alone are insufficient to form spatially useful dynamical coordinates in this setting. Removing history fitting also increases ADE and FDE, even though the three components remain linearly decodable. This shows that high probe scores alone do not guarantee a state that supports accurate rollout. Removing photometric consistency produces a smaller increase in the mean prediction errors; its effect on color sensitivity is evaluated directly in Panel (b). Photometric consistency reduces the mean color-induced drift of the current position readout from 0.44 to 0.24 pixels. This benefit does not extend consistently to future drift or FDE, so we interpret the objective as improving the color robustness of the current position-associated representation rather than as providing a stable gain in future prediction.

5 Discussion and Conclusion

GravJEPA learns structured motion states from video and couples them to a shared time-homogeneous transition, providing a common interface for prediction and latent-state intervention. By evolving states after observations stop, the architecture can extend motion predictions beyond the image boundary under the modeled dynamics, suggesting a path toward reasoning about object locations outside the field of view. Learning how actions modify velocity- or acceleration-associated states could further extend this interface to action-conditioned prediction. Richer dynamics and calibrated interventions are promising next steps toward physics-engine-like world models that connect visual perception, prediction, and interaction.

AI use statement. We used generative AI tools to assist with code development and manuscript polishing. The authors reviewed the AI-assisted content and verified the code, results, and references. We take full responsibility for the final content of this work.

REFERENCES

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15619–15629, 2023. doi: 10.1109/CVPR52729.2023.01499. URL https://openaccess.thecvf.com/content/CVPR2023/html/Assran_Self-Supervised_Learning_From_Images_With_a_Joint-Embedding_Predictive_Architecture_CVPR_2023_paper.html.

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. doi: 10.48550/arXiv.2506.09985. URL https://arxiv.org/abs/2506.09985.

Ilze Amanda Auzina, Çağatay Yıldız, Sara Magliacane, Matthias Bethge, and Efstratios Gavves. Modulated neural ODEs. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-1931. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/8bc74514d554a90c996576f6c373f5f3-Abstract-Conference.html.

Randall Balestriero and Yann LeCun. Lejepa: Provable and scalable self-supervised learning without the heuristics. *arXiv preprint arXiv:2511.08544*, 2025.

Hritik Bansal, Zongyu Lin, Tianyi Xie, Zeshun Zong, Michal Yarom, Yonatan Bitton, Chenfanfu Jiang, Yizhou Sun, Kai-Wei Chang, and Aditya Grover. Videophy: Evaluating physical commonsense for video generation. In *International Conference on Learning Representations*, volume 2025, pp. 102075–102121, 2025.

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. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=QaCCuDfBk2.

Kathleen Champion, Bethany Lusch, J Nathan Kutz, and Steven L Brunton. Data-driven discovery of coordinates and governing equations. *Proceedings of the National Academy of Sciences*, 116 (45):22445–22451, 2019.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K. Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, volume 31, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/69386f6bb1dfed68692a24c8686939b9-Abstract.html.

Emily L Denton and vighnesh Birodkar. Unsupervised learning of disentangled representations from video. *Advances in neural information processing systems*, 30, 2017.

Quentin Garrido, Laurent Najman, and Yann Lecun. Self-supervised learning of split invariant equivariant representations. *arXiv preprint arXiv:2302.10283*, 2023.

Quentin Garrido, Mahmoud Assran, Nicolas Ballas, Adrien Bardes, Laurent Najman, and Yann LeCun. Learning and leveraging world models in visual representation learning. *arXiv preprint arXiv:2403.00504*, 2024.

Vincent Le Guen and Nicolas Thome. Disentangling physical dynamics from unknown factors for unsupervised video prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11474–11484, 2020.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pp. 2555–2565. PMLR, 2019.

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://openreview.net/forum?id=S1lOTC4tDS.

Steeven Janny, Fabien Baradel, Natalia Neverova, Madiha Nadri, Greg Mori, and Christian Wolf. Filtered-cophy: Unsupervised learning of counterfactual physics in pixel space. *arXiv preprint arXiv:2202.00368*, 2022.

Miguel Jaques, Michael Burke, and Timothy Hospedales. Physics-as-inverse-graphics: Unsupervised physical parameter estimation from video. *arXiv preprint arXiv:1905.11169*, 2019.

Bingyi Kang, Yang Yue, Rui Lu, Zhijie Lin, Yang Zhao, Kaixin Wang, Gao Huang, and Jiashi Feng. How far is video generation from world model: A physical law perspective. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 28991–29017. PMLR, 2025. URL https://proceedings.mlr.press/v267/kang25g.html.

David Klindt, Yann LeCun, and Randall Balestriero. When does lejepa learn a world model? *arXiv preprint arXiv:2605.26379*, 2026.

Haodong Li, Shaoteng Liu, Tianyu Wang, Chongjian Ge, Sihui Ji, Jiahan Zhang, Xin Lin, Haolin Lu, Zhe Lin, and Manmohan Chandraker. Learning how the world evolves: Extrapolative video world models via latent dynamics reasoning. *arXiv preprint arXiv:2608.09926*, 2026a. doi: 10.48550/arXiv.2608.09926. URL https://arxiv.org/abs/2608.09926.

Jiaqi Li, Xinglong Zhang, Haibin Xie, Yixing Lan, Wei Pan, and Xin Xu. Koopman Dreamer: Spectrally constrained latent dynamics for stable world-model imagination. *arXiv preprint arXiv:2607.19719*, 2026b. doi: 10.48550/arXiv.2607.19719. URL https://arxiv.org/abs/2607.19719v2.

Kenneth Li, Aspen K Hopkins, David Bau, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Emergent world representations: Exploring a sequence model trained on a synthetic task. *arXiv preprint arXiv:2210.13382*, 2022.

Yingzhen Li and Stephan Mandt. Disentangled sequential autoencoder. *arXiv preprint arXiv:1803.02991*, 2018.

Andy Zeyi Liu, Haoran Sun, Lucas Baker, Randall Balestriero, and John Sous. Semigroup-JEPA: Latent dynamics consistency for zero-shot physics generalization, 2026a. URL https://arxiv.org/abs/2609.10464v1.

Dongxiu Liu, Haoyi Niu, Peng Cheng, Yuan Gao, Xirui Kang, Sangli Teng, Koushil Sreenath, and Xianyuan Zhan. ODEWorld: A continuous predictive architecture via physical-time flow. *arXiv preprint arXiv:2607.27924*, 2026b. doi: 10.48550/arXiv.2607.27924. URL https://arxiv.org/abs/2607.27924v3.

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, pp. 4114–4124. PMLR, 2019.

Francesco Locatello, Ben Poole, Gunnar Rätsch, Bernhard Schölkopf, Olivier Bachem, and Michael Tschannen. Weakly-supervised disentanglement without compromises. In *International conference on machine learning*, pp. 6348–6359. PMLR, 2020.

Bethany Lusch, J. Nathan Kutz, and Steven L. Brunton. Deep learning for universal linear embeddings of nonlinear dynamics. *Nature Communications*, 9(1):4950, 2018. doi: 10.1038/s41467-018-07210-0. URL https://www.nature.com/articles/s41467-018-07210-0.

Lucas Maes, Quentin Le Lidec, Damien Scieur, Yann LeCun, and Randall Balestriero. LeWorldModel: Stable end-to-end joint-embedding predictive architecture from pixels. *arXiv preprint arXiv:2603.19312*, 2026. doi: 10.48550/arXiv.2603.19312. URL https://arxiv.org/abs/2603.19312.

Saman Motamed, Laura Culp, Kevin Swersky, Priyank Jaini, and Robert Geirhos. Do generative video models understand physical principles? In *2026 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 948–958. IEEE, 2026.

Samuel E Otto and Clarence W Rowley. Linearly recurrent autoencoder networks for learning dynamics. *SIAM Journal on Applied Dynamical Systems*, 18(1):558–593, 2019.

Kaizhen Tan, Xin Xu, Siru Tao, Yixiao Li, Hanzhe Hong, Yang Feng, and Heqing Du. What can latent world models know? physical parameter identifiability in multimodal predictive representations. *arXiv preprint arXiv:2607.27017*, 2026. doi: 10.48550/arXiv.2607.27017. URL https://arxiv.org/abs/2607.27017v3.

Yuanyuan Wang, Wenjie Wang, Kun Zhang, and Mingming Gong. Physics from video: Identifiability of time-invariant second-order odes under minimal trajectory conditions. *arXiv preprint arXiv:2606.00115*, 2026.

Haiyu Wu, Randall Balestriero, and Morgan Levine. Visreg: Variance-invariance-sketching regularization for jepa training. *arXiv preprint arXiv:2606.02572*, 2026.

Ziyi Wu, Nikita Dvornik, Klaus Greff, Thomas Kipf, and Animesh Garg. Slotformer: Unsupervised visual dynamics simulation with object-centric models. *arXiv preprint arXiv:2210.05861*, 2022.

Haodong Yan, Jiaguan Zhu, Mingyuan Jia, Ruiqing Yin, Junjie He, Zhide Zhong, Junfeng Li, Jinxuan Lu, Hengtao Li, Tianran Zhang, Jiayi Chen, Wenxuan Song, Wen Chen, Yuxiang Gao, and Haoang Li. Is forward prediction enough? physical state grounding for JEPA world models. *arXiv preprint arXiv:2608.06799*, 2026. doi: 10.48550/arXiv.2608.06799. URL https://arxiv.org/abs/2608.06799.

Çağatay Yıldız, Markus Heinonen, and Harri Lähdesmäki. ODE2VAE: Deep generative second order ODEs with bayesian neural networks. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/99a401435dcb65c4008d3ad22c8cdad0-Paper.pdf.

Xi Zeng, Haojie Ren, and Ziying Song. Phylatent: Learning dynamics-relevant representations for jepa world models. *arXiv preprint arXiv:2608.05720*, 2026.

Zhenghao Zhang, Yuanxiang Wang, Zhenyu Guan, Yujia Yang, Bingkang Shi, Tianyu Zong, Hongzhu Yi, Guoqing Chao, Xingchen Chen, Tiankun Yang, et al. Delta-jepa: Learning action-sensitive world models via latent difference decoding. *arXiv preprint arXiv:2606.31232*, 2026.

Appendix

A Model implementation and training

A.1 Visual encoding, state estimation, and readout

The input is a sequence of 64×64 RGB images, normalized as $x/255 - 0.5$. The shared ViT uses 4×4 patches, width 32, two Transformer layers, four attention heads, and feed-forward width 64. Its 256 patch tokens and one class token yield a class-token representation that is projected to $D = 16$ dimensions. The first $d = 4$ coordinates form p; the remaining 12 coordinates are not used to infer v or g.

Seven observed frames define the relative history $\rho = \mathrm{vec}(p_1 - p_7, \ldots, p_7 - p_7)$. With $t_\star = t_7$ and $\mu = (t_7 - t_1)/6$, all timestamps are normalized as $\tau = (t - t_\star)/\mu$. The estimator combines this relative history with fixed timestamp features to produce two four-dimensional vectors, $v_\star$ and $g_\star$. The same μ is retained for every future query from this inferred state.

The core predicts

$$\widehat{p}(\tau) = p_\star + \tau v_\star + \tfrac{1}{2}\tau^2 g_\star, \qquad \widehat{z}_{\mathrm{core}}(\tau) = \big[\widehat{p}(\tau), R(\widehat{p}(\tau))\big], \tag{12}$$

where R returns 12 coordinates. The full prediction adds a memory-conditioned residual to these coordinates, while leaving the predicted p unchanged:

$$\widehat{z}(\tau; m) = \big[\widehat{p}(\tau), R(\widehat{p}(\tau)) + G(\widehat{p}(\tau), m)\big]. \tag{13}$$

Neither readout receives an additional future image. The memory code m is fixed after the observation cutoff.

A.2 Appearance memory and gradient separation

Memory is extracted from a single observed image. Specifically, the patch-embedding convolution produces 32-channel features before positional embeddings or Transformer mixing. Concatenating their spatial mean and population standard deviation gives a 64-dimensional statistic. A LayerNorm/MLP maps this statistic to a 16-dimensional memory code. The residual head G receives the concatenation of $\widehat{p} \in \mathbb{R}^4$ and $m \in \mathbb{R}^{16}$ and returns 12 coordinates. Its output layer is initialized to zero, so the initial residual does not alter the core prediction.

For each clip at each training update, two distinct observed-frame indices a, b are sampled. Each frame supplies a separate memory code for predicting the same six future targets. Let r_j^+ denote the last 12 coordinates of the encoded future frame, and let sg denote stop-gradient. The auxiliary objective is

$$\mathcal{L}_{\mathrm{mem}} = \frac{1}{2} \sum_{\ell \in \{a,b\}} \mathrm{MSE}\big(\mathrm{sg}\left(R(\widehat{p}_j)\right) + G\big(\mathrm{sg}(\widehat{p}_j), m_\ell\big), \mathrm{sg}(r_j^+)\big)\,, \tag{14}$$

where MSE averages over batch, future time, and feature coordinates. The patch statistics supplied to the memory encoder are detached as well. Consequently, this objective updates the memory encoder and residual head, but not the visual encoder, state estimator, or core readout. It does not include an additional agreement loss between m_a and m_b. At inference, only the final observed frame supplies memory.

The core and auxiliary branch are trained simultaneously from the first update, with independent optimizers and separate gradient boundaries. This is not a procedure that first trains and freezes the core and then fits memory. Gradients through the core's future representation targets remain enabled; there is no exponential-moving-average target encoder.

The history estimator's 84 inputs concatenate 28 relative-position coordinates and 56 timestamp features. For frame $i \in \{1, \ldots, 7\}$, the eight-dimensional time feature is

$$\eta_i = \big((\tau_i - (i-7))/6, 0, \ldots, 0\big). \tag{15}$$

The reference-grid subtraction makes this feature zero for a unit-spaced normalized history; it is an implementation choice, not a requirement of time-homogeneous dynamics. The estimator does not receive an additional physical-seconds input μ.

Table 5: Modules beyond the shared ViT. The nominal 16-output core reconstruction MLP contributes only its final 12 outputs; the first four predicted coordinates are the propagated p itself.

Module	Configuration
Projector	Linear $32 \to 64$, BatchNorm, GELU, Linear $64 \to 16$
History estimator	Linear $84 \to 64$, GELU, Linear $64 \to 8$
Core reconstruction	Linear $4 \to 32$, GELU, Linear $32 \to 16$
Memory encoder	LayerNorm(64), Linear $64 \to 64$, GELU, Linear $64 \to 16$
Memory residual	Linear $20 \to 64$, GELU, Linear $64 \to 64$, GELU, Linear $64 \to 12$

A.3 Training objectives and image transformations

The core objective is

$$\mathcal{L}_{\text{core}} = \mathcal{L}_{\text{pred}} + 0.25\mathcal{L}_{\text{hist}} + \mathcal{L}_{\text{photo}} + \mathcal{L}_{\text{eq}} + \mathcal{L}_{\text{VISReg}}. \tag{16}$$

Future prediction compares all 16 coordinates of the six predicted core embeddings with the corresponding shared-encoder targets. History fitting evaluates the same inferred state at all seven non-positive history offsets and compares its four position coordinates with the observed p history. Both losses average over batch, time, and feature dimensions. Core history and future targets are not detached. The auxiliary memory objective in Equation 14 has weight one and does not change the gradient of Equation 16.

Translation equivariance. For each clip, both Cartesian displacement components are sampled uniformly from $\{-4, -2, 2, 4\}$ input pixels. The entire seven-frame history receives the same translation, with four-pixel reflection padding. With positive d_y pointing upwards, the normalized displacement is $\delta = (d_x, d_y)/32$. Let $Q \in \mathbb{R}^{4\times 2}$ be the reduced-QR orthonormal factor of a learned matrix. If superscripts O and S denote original and spatially transformed views, the implemented objective is

$$\mathcal{L}_{\text{eq}} = \frac{1}{B}\sum_{b=1}^{B} \frac{\frac{1}{7d}\sum_{i=1}^{7} \|p_{bi}^{S} - p_{bi}^{O} - Q\delta_b\|_2^2}{\max\{\|Q\delta_b\|_2^2/d, 10^{-8}\}}, \qquad d = 4. \tag{17}$$

The denominator is not detached. Under exact orthonormality it equals $\|\delta_b\|_2^2/d$ before clamping. The transformation acts on the whole image and does not use object masks or physical-state labels.

Photometric consistency. One set of photometric parameters is shared across all 13 frames of a clip. Brightness and contrast are uniform in $[0.8, 1.2]$, saturation in $[0.6, 1.4]$, and each RGB channel gain in $[0.9, 1.1]$. The transform applies saturation relative to grayscale ($0.299R{+}0.587G{+}0.114B$), contrast around 0.5, then brightness and channel gain, and clips to $[0, 1]$. Writing C for this view,

$$\mathcal{L}_{\text{photo}} = \frac{1}{13Bd}\sum_{b=1}^{B}\sum_{i=1}^{13} \|p_{bi}^{C} - p_{bi}^{O}\|_2^2. \tag{18}$$

This is photometric consistency, not a background-replacement objective. No relative-history agreement loss, direct v/g agreement loss, slot-wise SIGReg, or dual-view prediction task is added.

Normalization context. Each transformation loss re-encodes its original and transformed views together in an auxiliary batch. The auxiliary projector BatchNorm uses batch statistics but does not update running buffers. Gradients are enabled for both members of each pair. Normal core history and target encodings update their training-time statistics. Memory features are captured from the existing history-encoder forward pass, without an additional encoder or BatchNorm call. The memory head itself contains LayerNorm rather than BatchNorm. Evaluation uses frozen running statistics.

A.4 Distribution regularization

VISReg (Wu et al., 2026) is applied in float32 to the full original 16-dimensional embeddings, arranged as $Z \in \mathbb{R}^{13\times B\times 16}$. For each time and feature coordinate, let

$$\bar{z}_{tk} = \frac{1}{B}\sum_b z_{tbk}, \qquad \sigma_{tk} = \max\left\{\sqrt{\frac{1}{B}\sum_b (z_{tbk}-\bar{z}_{tk})^2}, 10^{-6}\right\}. \tag{19}$$

The center and scale penalties are the means of $\bar{z}_{tk}^2$ and $(\sigma_{tk}-1)^2$, respectively. The shape penalty uses $(z_{tbk}-\bar{z}_{tk})/\operatorname{sg}(\sigma_{tk})$, projects onto 64 random unit directions, sorts each projection over the batch, and matches the sorted values to standard-normal quantiles $\Phi^{-1}(j/(B+1))$, $j=1,\dots,B$. The three penalties have unit weights. The standard deviation is detached only in the shape normalization, not in the center or scale penalties. Auxiliary views and inferred v/g slots are not separately regularized by VISReg.

A.5 Optimization and model identities

All three main comparison models are trained from scratch for 40,000 updates with batch size 256, using AdamW, weight decay 10^{-3}, gradient-norm clipping at 1 separately for core and memory, and bfloat16 training. We ran three seeds for each model. The peak learning rate is 3×10^{-4} for the dynamical core and AR models and 10^{-3} for the auxiliary memory head. A single cosine schedule spans the full run, with 1,000 warm-up updates and a final learning-rate factor of 0.1. Training used an NVIDIA H20 with Python 3.10.20, PyTorch 2.5.1+cu124, and CUDA 12.4. For zero-based update $n \in \{0,\dots,39999\}$, the implemented learning-rate multiplier is

$$\min\{1,(n+1)/1000\}\left[0.1+\frac{0.9}{2}\left(1+\cos\frac{\pi n}{39999}\right)\right]. \tag{20}$$

Training clips are sampled with replacement. Patch positional parameters are initialized on an 8×8 grid and interpolated to the 16×16 patch grid; this is random initialization, not the use of pretrained visual weights.

AR and AR+EIR predict complete 16-dimensional embeddings and are trained with six-step free-running rollout. AR+EIR applies the same transformation objectives to its first four coordinates, but does not have explicit $p/v/g$ slots. Table 6 counts all registered parameters; the non-encoder count is not solely predictor parameters. The comparison is not parameter- or computation-matched.

Table 6: Parameter counts for the formal comparison models.

Model	Encoder	Other components	Total
AR	26,976	38,480	65,456
AR+EIR	26,976	38,488	65,464
GravJEPA	26,976	23,208	50,184

B Controlled data generation

B.1 Physical trajectories and split construction

Each trajectory follows free constant-acceleration motion,

$$\mathbf{q}(t) = \mathbf{q}_\star + t\mathbf{v}_\star + \tfrac{1}{2}t^2\mathbf{g}, \tag{21}$$

with $t=0$ at the seventh observed frame. There are no collisions or boundary forces. The camera covers $[-2,2]^2$, corresponding to 16 input pixels per world-coordinate unit. Both coordinates of $\mathbf{q}_\star$ are sampled in $[-0.3,0.3]$; this is an anchor-position range, not a bound on positions throughout the trajectory.

Velocity is sampled using a uniform direction $\theta_v \in [0,2\pi)$ and a magnitude parameter $u \in [3.14, 3.24]$:

$$\mathbf{d}_v = (\cos\theta_v, \sin\theta_v), \qquad \mathbf{v}_\star = u\frac{\mathbf{d}_v}{\|\mathbf{d}_v\|_\infty}. \tag{22}$$

Thus, $u = \|\mathbf{v}_\star\|_\infty$, not the Euclidean speed. Gravity has independent uniform direction and Euclidean magnitude $\|\mathbf{g}\|_2 \in [6, 8]$. The velocity and gravity magnitude intervals are each divided into three equal-width bins, and sampling is balanced over the resulting 3×3 cells. Position, velocity, and gravity use independent random streams. This construction does not imply that positions at other times are statistically independent of velocity or gravity.

The training set contains 11,520 physical trajectories, instantiated as four clips per trajectory for 46,080 clips. Calibration and confirmation sets each contain 576 trajectories. No complete $(\mathbf{q}_\star, \mathbf{v}_\star, \mathbf{g})$ tuple is shared across splits. Marginal parameter ranges do overlap across these ID splits. The in-distribution generation uses prespecified ranges that keep all 13 frames visible, rather than per-trajectory rejection or recentering.

Figure 4 visualizes the saved training parameters, counting each physical trajectory once. The square-shaped velocity support follows from sampling the infinity norm in Eq. 22; gravity instead occupies an annulus because its Euclidean norm is sampled. These are distributions at the observation cutoff, not distributions pooled over all video frames.

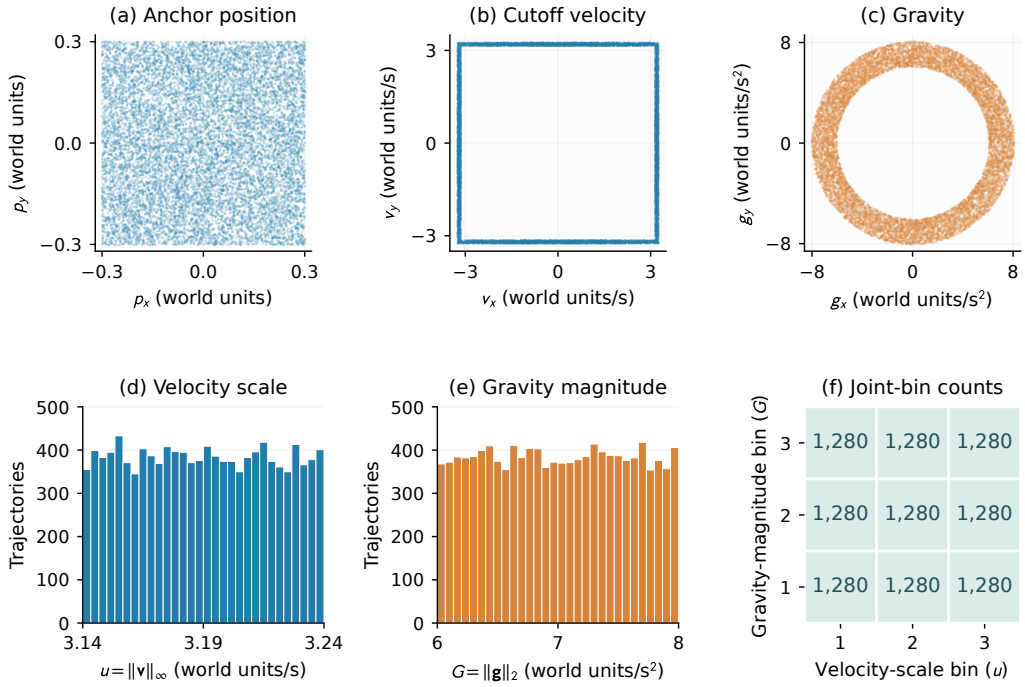


Figure 4: **Physical-parameter sampling in the training set.** The saved parameters of all 11,520 physical trajectories are shown once, without counting their four clip realizations as independent trajectories. The upper panels show cutoff position, velocity, and gravity in their two-dimensional coordinate spaces. The lower panels show the sampled velocity magnitude $u = \|\mathbf{v}_\star\|_\infty$, gravity magnitude $G = \|\mathbf{g}\|_2$, and the counts in their 3×3 sampling grid. Position, velocity, and gravity use world units, world units/s, and world units/s^2, respectively. Each magnitude cell contains 1,280 trajectories. The square-shaped velocity support and annular gravity support reflect different norm conventions, not clipping or boundary interactions.

B.2 Physical-variable sampling correlations

We audit Pearson correlations between the ground-truth cutoff position $\mathbf{q}_\star$, cutoff velocity $\mathbf{v}_\star$, and constant gravity $\mathbf{g}$, using each saved physical trajectory once. Table 7 reports the signed training-set correlation matrix and a cross-factor summary for each split. For two distinct vector-valued factors

$\mathbf{a}$ and $\mathbf{b}$, the summary statistic is

$$R_{ab}^{\max} = \max_{i,j\in\{x,y\}} |\mathrm{Corr}(a_i, b_j)| \,. \tag{23}$$

These summaries are computed from the saved 6×6 matrices; they are not correlations between learned latent components.

Table 7: **Correlation audit of the physical sampling.** (a) Signed Pearson correlations for the 11,520 training trajectories. (b) Maximum absolute cross-coordinate correlations between each pair of physical factors, computed separately within each split. Position q and velocity v refer to the observation cutoff. Each trajectory contributes once, not once per clip or frame. Values are rounded to four decimal places.

(a) Training-set correlation matrix

	$q_{\star,x}$	$q_{\star,y}$	$v_{\star,x}$	$v_{\star,y}$	g_x	g_y
$q_{\star,x}$	1.0000	0.0183	-0.0068	0.0017	-0.0008	0.0122
$q_{\star,y}$	0.0183	1.0000	0.0020	-0.0019	0.0022	0.0020
$v_{\star,x}$	-0.0068	0.0020	1.0000	0.0138	-0.0036	0.0104
$v_{\star,y}$	0.0017	-0.0019	0.0138	1.0000	0.0062	-0.0068
g_x	-0.0008	0.0022	-0.0036	0.0062	1.0000	0.0002
g_y	0.0122	0.0020	0.0104	-0.0068	0.0002	1.0000

(b) Cross-factor summary by split

Split	Trajectories	$R_{qv}^{\max}$	$R_{qg}^{\max}$	$R_{vg}^{\max}$
Training	11,520	0.0068	0.0122	0.0104
Calibration	576	0.1039	0.0273	0.0905
Confirmation	576	0.0480	0.0588	0.0411

The largest absolute cross-factor correlation in the training sample is 0.0122. These small linear correlations characterize the realized sample; they do not establish statistical independence, exclude nonlinear dependencies, or demonstrate latent disentanglement.

B.3 Appearance and rendering

Each clip contains one circle, square, or equilateral triangle, with circumradius in $\{0.12, 0.15, 0.18\}$. Equal circumradius does not imply equal foreground area. Eight colors give 72 possible shape–color–size combinations. Appearance is fixed within a clip; it is not resampled independently at each frame.

The original RGB palette is

$$(205, 45, 45),\ (40, 85, 205),\ (35, 145, 75),\ (135, 60, 180),$$
$$(200, 105, 25),\ (25, 145, 155),\ (185, 55, 130),\ (45, 45, 45).$$

Rendering first uses a gray background of 238 and 128×128 subpixels, followed by 2×2 averaging to 64×64. The affine image transform $I \mapsto 127.5 + 0.5(I - 238)$ is then applied to the entire image, so the actual background is 127.5 and the object colors are transformed as well. Metadata background identifiers do not produce background-content variation in this benchmark.

B.4 Observation and query times

Let $a_1, \ldots, a_6$ be positive history-gap weights and let μ denote the mean physical interval. The six gaps are $\Delta t_i = \mu a_i / (\frac{1}{6}\sum_{k=1}^{6} a_k)$, and history timestamps are accumulated backwards from the anchor at zero. During training, approximately 20% of clips use uniform 0.055-second gaps; the remaining clips use $a_i \sim U(0.5, 1.5)$ and $\mu \sim U(0.044, 0.055)$ seconds. Future training targets are always at $0.055, 0.110, \ldots, 0.330$ seconds.

C READOUTS AND METRICS

C.1 FROZEN PHYSICAL-VARIABLE READOUTS

World-model training does not use physical-state targets. Physical labels are used after training to fit diagnostic readouts on calibration data. The main full-embedding and p-only position readouts use the $576 \times 13 = 7,488$ encoded calibration frames. Independent velocity and gravity readouts use the 576 inferred histories, with features v/μ and g/μ^2, respectively, to account for physical time units.

For calibration features X, each feature is centered and divided by its sample standard deviation, clamped below at 10^{-5}, to give $\widetilde{X}$. Targets Y are centered but not rescaled. The fixed ridge coefficient is $\lambda = 10^{-3}$, without test-set selection:

$$W = \left(\frac{\widetilde{X}^\top \widetilde{X}}{n} + \lambda I \right)^{-1} \frac{\widetilde{X}^\top (Y - \overline{Y})}{n}, \qquad \widehat{Y} = \widetilde{X} W + \overline{Y}. \tag{24}$$

The solve uses double precision. Feature statistics, coefficients, and intercepts are frozen across confirmation conditions. Independent slot probes may learn different coordinate maps. A separate common-p diagnostic applies the position readout's linear map to v/μ and g/μ^2, without the position intercept; it is not interchangeable with independently fitted slot probes. The position-feature mean is not subtracted in this cross-slot diagnostic; only feature scaling and the linear coefficients are used.

C.2 METRICS AND UNCERTAINTY

For N trajectories and J query times, the main position metrics are

$$\mathrm{ADE} = \frac{16}{NJ} \sum_{i=1}^{N} \sum_{j=1}^{J} \|\widehat{\mathbf{q}}_{ij} - \mathbf{q}_{ij}\|_2, \qquad \mathrm{FDE} = \frac{16}{N} \sum_{i=1}^{N} \|\widehat{\mathbf{q}}_{iJ} - \mathbf{q}_{iJ}\|_2. \tag{25}$$

The factor 16 converts world-coordinate errors to input pixels. For Table 1(a), we report the mean and standard deviation of ADE and FDE across the three independently trained seeds. Physical-variable R^2 in the main comparison pools squared error and centered target variation across samples and output coordinates. The appearance-probe diagnostic below instead uses the mean of the two coordinate-wise position R^2 values. Position probing of observed embeddings is not an R^2 score for future prediction.

Where confidence intervals are reported, paired bootstrap resampling uses 2,000 draws of physical trajectories and percentile 95% intervals. All conditions or appearance variants belonging to a trajectory remain together. These intervals describe variation over the evaluated trajectories conditional on a fixed trained model; they are not cross-training-seed uncertainty. They are not adjusted for multiple comparisons. The confirmation pools have been inspected in previous diagnostics and are not newly held-out blind tests.

D PHYSICAL MAGNITUDE GENERALIZATION

The ID supports are $u = \|\mathbf{v}_\star\|_\infty \in [3.14, 3.24]$ and $G = \|\mathbf{g}\|_2 \in [6, 8]$. Low/high velocity conditions use $[2.0, 2.8]$ and $[3.6, 4.5]$, while low/high gravity conditions use $[3, 5]$ and $[9, 12]$. Joint OOD sets both factors to their high ranges. Within each factor, a shared unit-interval quantile ξ is mapped to the corresponding interval as $l + (h - l)\xi$. Cutoff position, vector directions, appearance, timestamps, and quantiles are paired across conditions. Unshifted factors retain their ID values.

The paired pool starts from 576 candidates generated with seed 9252601. All models use a common subset of 536 histories. A candidate is retained only if, under every physical condition, all seven observed centers satisfy $\max_{j \le 7} \|\mathbf{q}(t_j)\|_\infty + 0.18 \le 2$. The radius 0.18 is a conservative bound shared across object sizes. Future visibility is not used for filtering. The same frozen readout is used for six queries over $H_{\mathrm{ref}} = 0.33$ seconds.

Figure 5 shows the resulting paired distributions relative to the training support. The visibility intersection conditions the candidate pool, so the retained factors are not guaranteed to remain independent or exactly uniform.

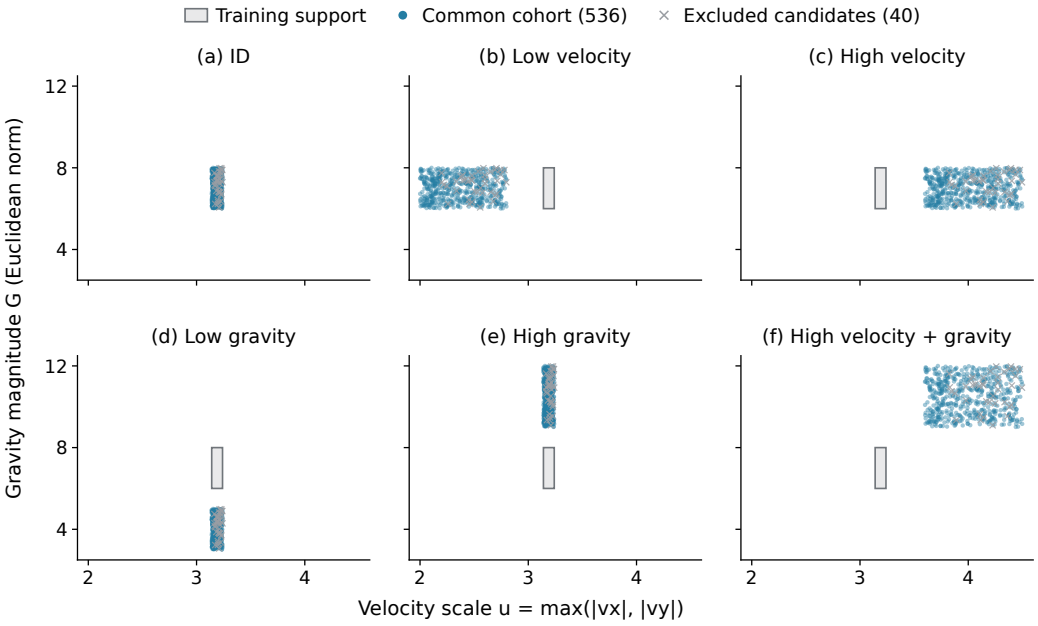


Figure 5: **Sampling distributions for physical-OOD evaluation.** The axes show velocity magnitude $u = \|\mathbf{v}_\star\|_\infty$ and gravity magnitude $G = \|\mathbf{g}\|_2$, in world units/s and world units/s^2, respectively. The shaded rectangle marks the prescribed training support. Each panel maps the same 576 candidate contexts to one physical condition. Blue points denote the 536 contexts whose complete observation histories are visible in all six conditions; gray crosses denote the remaining 40 candidates. Exclusion is determined by this cross-condition intersection, not necessarily by visibility in the displayed panel. No future-visibility filtering is applied.

Table 8: Full physical-OOD results, as ADE/FDE in pixels on 536 paired histories. Paired ID is evaluated on this common pool, not on the 576-trajectory main test set.

Condition	AR	AR+EIR	GravJEPA
Paired ID	11.86 / 19.96	3.37 / 5.60	1.99 / 4.31
Low velocity	9.28 / 15.35	3.73 / 6.07	1.80 / 3.81
High velocity	14.57 / 24.71	4.44 / 8.28	2.19 / 4.81
Low gravity	11.68 / 19.45	3.20 / 5.08	1.87 / 4.11
High gravity	12.16 / 20.80	3.61 / 6.48	2.23 / 4.71
High velocity + high gravity	14.82 / 25.42	4.61 / 8.98	2.47 / 5.31

The main-text velocity and gravity summaries weight the corresponding low and high conditions equally. Smaller magnitudes can reduce future displacement and need not be harder than ID. These results concern the declared magnitude shifts, not unseen directions or accurate identification of physical gravity.

E TEMPORAL REUSE, LONG-HORIZON QUERIES, AND COMPUTATION

E.1 FIXED-STATE COMPOSITION AND RE-ESTIMATION

For a fixed inferred state and a fixed normalized clock, the prescribed generator satisfies $A^3 = 0$. Thus,

$$K_\tau = e^{A\tau} = I + \tau A + \tfrac{1}{2}\tau^2 A^2, \qquad K_{\tau_2} K_{\tau_1} = K_{\tau_1+\tau_2}. \tag{26}$$

The implementation evaluates the equivalent constant-acceleration polynomial directly; it does not require a general matrix-exponential routine. With memory fixed, the deterministic readout gives the same future output for direct and composed evolution up to numerical error. Across the evaluated cases, decoded positions from direct and composed evolution differed by less than 2×10^{-5} input pixels. This is an implementation-level consistency check of the prescribed transition, not evidence of long-horizon accuracy or agreement with a state newly inferred after re-observation. This is a structural property. An initial state error still produces $\delta p(\tau) = \delta p_\star + \tau \delta v_\star + \frac{1}{2}\tau^2 \delta g_\star$.

Re-estimation after new observations is a different test. We compare an early history at frame indices $[-6, \ldots, 0]$ with a later history $[k-6, \ldots, k]$, for $k \in \{1, 3\}$, using 0.055-second intervals. Both paths predict the common absolute future indices $[k+1, \ldots, k+6]$. The early path continues without new observations; the later path estimates a new state from its seven real frames. Disagreement averages the distance between their six predictions; both FDE values use the common final physical time. All 576 diagnostic trajectories pass the observed-window visibility check. Future targets are retained even after leaving the view.

Table 9: Changing the observation cutoff. All errors are in pixels. The later path has additional evidence and a shorter prediction offset, so this is not an equal-information comparison.

Shift k	Model	Disagreement	Continued FDE	Re-estimated FDE
1	AR	0.012	22.869	22.869
1	AR+EIR	1.897	6.918	6.254
1	GravJEPA	1.743	4.802	3.831
3	AR	0.023	30.091	30.091
3	AR+EIR	4.018	13.868	12.683
3	GravJEPA	4.026	8.124	4.021

GravJEPA has lower continued and re-estimated FDE in this test, but does not have lower disagreement than AR+EIR at $k = 3$. Small disagreement alone is not evidence of accurate dynamics. This test also changes the anchor-state distribution: at $k = 3$, 92.0% of positions and 96.0% of velocity maximum components lie outside the training anchor ranges. If a new history uses a different μ', state comparison additionally requires the clock conversions $v' = (\mu'/\mu)v$ and $g' = (\mu'/\mu)^2 g$.

E.2 INDEPENDENT FUTURE ENDPOINT QUERIES

Each request returns one complete future embedding and its decoded position at $H \times 0.055$ seconds for $H \in \{3, 6, 9, 12, 18, 24, 36, 48\}$. The seven observed images and their time normalization are fixed. GravJEPA queries the same inferred state and memory. Unit-step AR performs the intervening latent updates. Query-step AR starts independently from the original encoded history for each H and makes one update conditioned on the requested offset; it does not chain outputs from different endpoint requests.

Ground truth is the analytic extension of the same free-flight trajectory, without collisions, coordinate clipping, or future-visibility filtering. No future images are used. This 576-trajectory diagnostic pool differs from the main confirmation protocol; its H6 error is therefore not expected to reproduce the main table exactly.

Only 9.0% of target centers remain in view at H24, and none at H36 or H48. GravJEPA's lower relative error does not imply uniformly accurate extrapolation: its H48 FDE is 273.107 pixels.

Table 10: Single-endpoint query FDE in pixels on the fixed diagnostic pool. H6 is the training prediction horizon; longer horizons also include increasing numbers of out-of-view targets.

		AR+EIR		AR	
H	GravJEPA	Unit-step	Query-step	Unit-step	Query-step
3	1.214	3.008	5.202	9.719	9.719
6	3.537	5.105	13.290	19.431	19.431
9	8.124	13.868	22.731	30.091	30.091
12	15.016	35.225	34.496	42.015	42.015
18	35.591	86.028	66.689	71.926	71.926
24	65.205	122.262	110.674	113.858	113.859
36	151.258	230.743	234.443	235.841	235.843
48	273.107	410.257	406.207	407.018	407.020

E.3 Matrix-operation accounting

We count the main matrix multiply–accumulate operations (MACs) in the actual batch-one forward paths: Linear and Conv2d operations, attention $QK^\top$ and AV, and the frozen 16×2 position readout. Attention is counted as dense computation. Bias, normalization, softmax, activations, elementwise modulation and state arithmetic, pooling, copying, and memory traffic are excluded. These are neither exhaustive FLOP counts nor measured hardware latency.

Table 11: Counted matrix MACs for one requested endpoint. Initialization encodes the seven-frame history and, for GravJEPA, estimates state and memory. No future predictions are cached. Here H denotes the requested horizon in 0.055-second AR steps.

Component or policy	GravJEPA	AR / AR+EIR
Initialization	91,439,744	91,428,736
Core latent readout per query	640	—
Memory residual per query	6,144	—
One AR update	—	242,816
Position readout per output	32	32
Cached single endpoint	6,816	$242,816H + 32$ (unit-step)
Cached single endpoint	6,816	242,848 (query-step)

The reported ratio is $242,848/6,816 \simeq 35.63$ for a cached single query. Both direct GravJEPA queries and query-step AR have horizon-independent cost for a fixed number of outputs. For Q requested outputs, readout costs scale with Q; the entire workload is not constant-cost. Including initialization gives 91.446560 versus 91.671584 million MACs for one GravJEPA versus query-step AR request, so the cached-query ratio is not an end-to-end speedup. The AR implementation processes a seven-latent context at each update and does not introduce an additional key/value caching optimization.

E.4 Out-of-view reporting

Long-horizon main-table evaluation retains all 576 confirmation trajectories and uses the underlying physical coordinates after exit. The reported exit counts imply 504 true positives, 8 false negatives, 60 false positives, and 4 true negatives for GravJEPA. These counts should be considered together with coordinate error, rather than interpreting a high true-positive count alone as reliable exit detection.

F Representation analysis and state interventions

F.1 Color and shape readouts

We render 576 physical contexts in seven chromatic training colors or in all three training shapes. Within each context, other physical and visual factors are held fixed. The color diagnostic excludes

the eighth dark-gray palette entry; it is not a held-out-color test. Shape variants share circumradius, not foreground area.

Contexts are split into 384/96/96 training/validation/test groups using split seed 9252801, and every rendering of a context stays in the same group. Features are extracted at the observation anchor: p_4, the remaining 12 coordinates, the full 16-dimensional embedding, and the 16-dimensional memory code for GravJEPA. Here memory is computed from that same anchor image's early patch statistics. AR+EIR's first four coordinates are EIR-regularized coordinates, not an inferred $p/v/g$ decomposition.

All diagnostic features are standardized using probe-training statistics only, with standard deviations clamped below at 10^{-6}. Linear classification uses logistic regression with $C = 1$ and a maximum of 3,000 iterations. Nonlinear classification uses hidden widths 64 and 32 with GELU, cross-entropy, AdamW at learning rate 10^{-3} and weight decay 10^{-4}, batch size 256, and 1,000 updates. Validation cross-entropy is checked every 25 updates to select a checkpoint. Three probe initializations are averaged; the world model remains the same frozen model seed. Position readout for these diagnostics is a separate ridge probe with coefficient 1, not the main forecasting probe in Equation 24.

Chance accuracy is $1/7$ for color and $1/3$ for shape. Context-level bootstrap preserves all variants of a context. Unequal feature dimensions and shape-dependent area should be considered when comparing components. These probes measure information accessibility; they do not demonstrate appearance invariance, complete disentanglement, or that the predictor uses every decodable attribute.

Table 3 shows that GravJEPA's memory carries substantially more color information than p, whereas shape allocation is less distinct. The p component remains strongly predictive of position ($R^2 = 0.967/0.971$ on the color/shape sets) but is still appearance-readable. AR+EIR has no explicit $p/v/g$ factorization, yet its full embedding supports both position readout ($R^2 = 0.996/0.995$) and appearance classification. Overall, the results indicate partial information allocation rather than complete disentanglement or appearance invariance.

F.2 Controlled state grids

The organization diagnostic varies position, velocity, or gravity over an 11×11 physical grid while fixing the other two quantities at the observation cutoff, along with appearance and observation times. The corresponding latent components are projected separately. Projection axes, scales, or areas should not be compared across independently fitted t-SNE plots, and a visually smooth grid is not itself a physical-identification test.

F.3 Matched-donor and direct latent-state interventions

Frozen checkpoint and cohort. The evaluation uses a GravJEPA checkpoint. Model parameters, normalization statistics, and the original calibration-fitted position readouts are frozen. No model optimization, probe fitting, or latent-coordinate calibration uses intervention examples. This checkpoint was trained with seven observed frames and six future targets, not with H12 or H24 training targets.

Starting from the 576 irregular-ID confirmation trajectories, we retain the common subset for which all eight counterfactual clips (four velocity edits and four gravity edits) remain fully visible at the original 13 frame times. The geometric criterion is $\|\mathbf{q}(t)\|_\infty + r \le 2 - 1/32$, using the object's rendering radius $r \in \{0.12, 0.15, 0.18\}$. This gives $N = 344$ trajectories. For H12 and H24, all 344 trajectories are retained, including futures that leave the view. Longer-horizon errors are computed from physical coordinates, without encoding blank future images as targets.

Matched component replacement. For each source trajectory, donor histories are generated by zeroing, reversing, rotating counterclockwise by $90°$, or scaling by 1.5 either physical velocity or gravity. Cutoff position, the other physical vector, appearance, and observation timestamps are fixed. Only the donor's seven observed frames are encoded.

The matched single-slot intervention replaces the corresponding inferred v or g component and retains the source p, the other motion component, and the source last-frame memory. It does not

replace the full inferred state. The matched both-slot control takes both v and g from the same donor while retaining source p and memory. The shuffled-single control cyclically shifts donor identities by one trajectory and replaces only the designated component.

Direct latent arithmetic. We separately multiply the source latent component by 0, -1, or 1.5, without constructing a donor. A raw 90° rotation is not defined, since the four-dimensional latent has no prescribed two-dimensional physical coordinate plane. In particular, replacing g with the inferred component from a zero-gravity donor does not assume that the latent origin represents physical zero gravity; setting the original g vector to zero does. The two tests therefore make different demands on the representation.

Queries and readouts. Queries are spaced by 0.055 seconds, with endpoints H6, H12, and H24 at 0.33, 0.66, and 1.32 seconds. The source history's original μ is retained throughout, and the primary readout uses the full 16-dimensional prediction with fixed source memory. Predicted trajectories are not translated onto the true anchor. The stored diagnostics also include a p-only readout and a memory-disabled core readout. The latter retains the original full position probe; it is a readout ablation, not an independently refitted optimal core probe.

Response metrics. Let $\widehat{\mathbf{q}}_i(t)$ and $\widehat{\mathbf{q}}_i^e(t)$ denote the original and edited predictions. The target and predicted displacements caused by the edit are

$$\mathbf{d}_i(t) = t(\mathbf{v}_i^e - \mathbf{v}_i) + \tfrac{1}{2}t^2(\mathbf{g}_i^e - \mathbf{g}_i), \qquad \widehat{\mathbf{d}}_i(t) = \widehat{\mathbf{q}}_i^e(t) - \widehat{\mathbf{q}}_i(t). \tag{27}$$

At the endpoint T, we report the mean response-error ratio and response gain:

$$\mathcal{E}_{\mathrm{resp}}(T) = \frac{1}{N}\sum_i \frac{\|\widehat{\mathbf{d}}_i(T) - \mathbf{d}_i(T)\|_2^2}{\|\mathbf{d}_i(T)\|_2^2}, \qquad \gamma(T) = \frac{1}{N}\sum_i \frac{\widehat{\mathbf{d}}_i(T)^\top \mathbf{d}_i(T)}{\|\mathbf{d}_i(T)\|_2^2}. \tag{28}$$

An exact response gives $\mathcal{E}_{\mathrm{resp}} = 0$ and $\gamma = 1$; ignoring the edit gives $\mathcal{E}_{\mathrm{resp}} = 1$. The error ratio is not an accuracy, success rate, or R^2 score. The implementation asserts $\|\mathbf{d}_i(T)\|_2^2 > 10^{-9}$ in squared world-coordinate units for every evaluated example. No denominator smoothing, clipping, or denominator-based sample removal is applied.

Counterfactual ADE and FDE use the target $\mathbf{q}_i^e(t) = \mathbf{q}_i(t) + \mathbf{d}_i(t)$ and the same 16-pixel conversion as the main metrics. To measure the benefit of an edit, edited and unedited predictions must both be scored against this counterfactual target, rather than against different futures. Bootstrap intervals use 2,000 paired trajectory resamples with seed 9265101. For operation-averaged summaries, operations are first averaged within each trajectory; operations and time points are not treated as independent samples.

Table 12: H6 interventions on the formal frozen checkpoint. Each row uses the same 344 trajectories and full-memory readout. $\mathcal{E}$ is the response-error ratio, γ is response gain, and FDE is counterfactual endpoint error in pixels. A dash denotes an undefined raw rotation, not a missing experimental run.

		Matched single-slot			Direct latent arithmetic		
Slot	Edit	$\mathcal{E}\downarrow$	γ	FDE↓	$\mathcal{E}\downarrow$	γ	FDE↓
v	Zero	.022	1.101	5.326	.029	1.113	5.063
v	Reverse	.016	1.096	7.023	.029	1.112	7.065
v	Rotate 90°	.018	1.100	5.622	–	–	–
v	Scale 1.5	.047	1.125	3.396	.031	1.113	4.055
g	Zero	.109	.893	3.834	.776	1.071	2.915
g	Reverse	.037	.901	3.816	.780	1.072	7.142
g	Rotate 90°	.059	.891	3.807	–	–	–
g	Scale 1.5	.419	.942	3.883	.772	1.070	6.208

Long-horizon behavior and scope. Matched response ratios vary little from H6 to H24, but absolute trajectory errors grow substantially. The unedited model's FDE against the original trajectory

is 3.945, 17.246, and 74.652 pixels at H6, H12, and H24. These original-task errors and the counterfactual FDE values have different targets and are not paired intervention benefits. Stable relative responses do not establish accurate long-horizon trajectories.

The prescribed flow determines the temporal form of a latent edit's effect. The experiments test whether visually inferred components produce the appropriate physical responses through that interface, not whether the network discovers an unknown evolution law. Donors require simulator-controlled histories and are used only for evaluation. Zeroing and scaling may also leave the training support.

G Training Configuration for Objective Ablations

The ablation is conducted on a small-scale configuration useing 1,152 training trajectories and 4,608 clips, 4,000 updates, and batch size 64. Its two-cycle cosine schedule differs from the single 40,000-update schedule described in Appendix A.5.