DO-CAUSAL-JEPA: LEARNING INTERVENTIONAL WORLD MODELS WITH GRAPH SURGERY ESTIMATOR

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent advances on Joint Predictive Embedding Architecture (JEPA) have shown significant progress in learning object-centric representations and the dynamics of these representations from images and videos. One prominent model is known as Causal-JEPA (C-JEPA) (Nam et al., 2026), which learns the interaction dynamics of objects by masking different feature maps based on images from previous time frames to predict the next time frame's representation state. The model is shown to be able to learn an important notion called *influence neighborhoods*, which are claimed to be the minimal predictor sets of each object in the image without causal sufficiency under mild assumptions. However, the relationship between influence neighborhoods and some classical causality concepts, such as invariant causal predictors, is not well-understood. In this work, we show that C-JEPA does not learn invariant causal predictors in a controlled environment in the presence of latent confounders. We propose an alternative training scheme to introduce an additional loss term that is based on graph surgery operations for learning object-level invariant predictors given a causal structure. Empirically, our proposed algorithm Do-CJEPA outperforms C-JEPA in learning the dynamics of the objects under mechanism shifts by having a smaller out-of-distribution test error gap and prediction errors in terms of mean squared error for both one-step prediction and rollout under shifted test environments.

1 INTRODUCTION

World models (Ha & Schmidhuber, 2018) aim to learn latent representations of dynamic environments that can support reasoning, planning, and control. A key feature for effective world modeling is to learn representations that explicitly capture objects and the interactions among them, rather than depending on potentially spurious correlations at the pixel level (Zholus et al.; Nishimoto & Matsubara, 2026; Ferraro et al., 2025). Object-centric representations provide one promising abstraction. For example, instead of representing a frame as a dense grid of pixels or patches, an encoder maps it to a set of slots, where each slot is intended to bind to an entity in the scene (Locatello et al., 2020). Object-centric models have been widely explored for modeling visual dynamics and have increasingly served as a foundation for learning world models (Wu et al., 2022; Villar-Corrales et al., 2023; Kipf et al., 2021; Zadaianchuk et al., 2023). Nevertheless, object-centric representations alone are insufficient for world modeling due to the lack of explicit mechanism to learn the interactions among objects. Some existing work explicitly separates the modeling of an object's temporal dynamics from its interactions with other objects (Feng et al., 2026). For instance, Lei et al. (2026) regularize attention so that each object attends to only a small subset of other objects. Kipf et al. (2021) represent object relations using an explicit graph structure.

Recently, Causal-JEPA (C-JEPA) extends joint embedding predictive architectures to object-centric world modeling by masking object-level latent trajectories rather than image patches (Nam et al., 2026). More specifically, it explicitly models object interactions by introducing a causal inductive bias through object-level masking interventions that control what information is observable to the predictor. Given slots extracted by a frozen object-centric encoder, C-JEPA uniformly samples object indices to mask. It preserves an initial identity anchor for masked objects and trains a transformer predictor to recover masked history and future latent states. This objective significantly improves interaction-dependent prediction. The model cannot simply copy the masked object's own trajectory and must infer it from surrounding objects and auxiliary variables. However, object-level

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.

masking alone does not guarantee invariance to mechanism shifts. A masked-slot predictor may still suffer from distribution shifts due to unobserved confounding when the slots fail to represent some context variables generated in the source environment. If one object-state mechanism changes at test time, the conditional distribution used by the predictor can change even when the physical relation of interest remains stable.

We propose Do-CJEPA, which incorporates the causal structure of the slots in the form of acyclic directed mixed graphs to restrict the self-attention in the transformer of C-JEPA for object-centric world modeling under mechanism shifts. Instead of randomly masking latent object trajectories, Do-CJEPA learns to construct predictors that are invariant across target domains using distributional invariances via Pearl's do-calculus (Pearl, 2009). Do-CJEPA models mechanism shifts as an intervention on the observed variables and performs a *graph surgery* to sever the dependency of the shifts on the predictive target by penalizing predictive error from the mean of the interventional distribution that can be estimated from source-environment data alone according to the causal structure. Do-CJEPA does not require the knowledge of when the mechanism shifts happen. This work is motivated by leveraging causal knowledge of the factors that generate images to construct invariant predictors (Peters et al., 2016; Kocaoglu et al., 2018; Subbaswamy et al., 2019; Gamella & Heinze-Deml, 2020; Jaber et al., 2020; Lee et al., 2023). Based on the experiment with the Interventional Pong dataset (Lippe et al., 2022), Do-CJEPA reduced the final-frame prediction mean squared error for object slots that inherit the mechanism shifts from the previous time frame by 69% relative to C-JEPA. The increase in this error between normal and intervened test trajectories was also 90% smaller. When the causal structure was estimated rather than given, Do-CJEPA retained most of the benefit, achieving a 68% reduction in shifted invariant error and finishing within 4% of the known-graph model. Our contributions are summarized below:

- We show how C-JEPA fails to produce invariant prediction in the presence of latent confounders under mechanism shifts. We propose Do-CJEPA that learns invariant predictors in the presence of latent confounders under mechanism shifts with causal knowledge about the generating factors of the images.
- We provide a version of Do-CJEPA that does not require any prior knowledge of the generating factors of images for learning invariant predictors through causal representation learning and causal discovery.
- We empirically show Do-CJEPA outperforms C-JEPA in next-state representation prediction with object-centric representations. We show that Do-JEPA can outperform C-JEPA even when the causal knowledge is imperfect.

2 Background

2.1 Causal Graphical Models

A **structural causal model** (SCM) is a tuple $\mathcal{M} = (U, V, \mathcal{F}, P(U))$, where U is a set of exogenous variables, $V = \{V_1, \ldots, V_d\}$ is a set of endogenous variables, $P(\mathbf{U})$ is a distribution over exogenous variables, and $\mathcal{F} = \{f_1, \ldots, f_d\}$ is a collection of structural assignments $V_i := f_i(\mathrm{pa}(V_i), U_i)$, where $\mathrm{pa}(V_i) \subseteq V \setminus \{V_i\}$ denotes the observed parents of V_i (Pearl, 2009). The causal graph G associated with an SCM contains a directed edge $V_j \to V_i$ whenever V_j appears as an argument of f_i. Unobserved common causes are either explicitly as latent variables or, after latent projection, as bidirected edges in an acyclic directed mixed graph (ADMG). An intervention $do(X = x)$ replaces the structural equations for variables $X \subseteq V$ by constants x. Graphically, this operation removes all incoming edges into X, denoted as $G_{\overline{X}}$. The post-intervention distribution is denoted by $P(V \setminus X \mid do(X = x))$ or equivalently by $P_x(\mathbf{V} \setminus X)$.

For a set $Y \subseteq V$, its *parents* are $\mathrm{pa}_G(Y) = \{X \in V : \text{there exists } J \in Y \text{ such that } X \to J \text{ in } G\}$. Its *ancestors*, denoted $\mathrm{An}_G(Y)$, consist of Y and every variable X for which a directed path from X to some $J \in Y$ exists in G. A path in an ADMG is a sequence of repeated adjacent vertices connected by directed or bidirected edges. For a non-endpoint vertex W on a path, we say that W is a *collider* on the path if both adjacent edges on the path have arrowheads pointing into W. If W is not a collider on the path, then W is called a *non-collider*. Let $Z \subseteq V$ be a conditioning set. A path is said to be *m-connecting* given Z if every non-collider on the path is not in Z, and every collider on the path is in $An_G(Z)$ where $An_G(Z)$ denotes the set of observed ancestors of

variables in Z through directed paths in G. If either condition fails, the path is *blocked* given Z. Two disjoint sets of variables $X, Y \subseteq V$ are *m-separated* by Z, denoted $(X \perp\!\!\!\perp Y \mid Z)_G$ if there is no m-connecting path between any variable in X and any variable in Y given Z in G. Otherwise, X and Y are m-connecting given Z, denoted as $(X \not\perp\!\!\!\perp Y \mid Z)_G$. A **c-component** is a maximal set of vertices connected by bidirected paths. We write $C_G(Y)$ for the c-component containing Y in G. For $A \subseteq V$, let $G[A]$ denote the subgraph induced by A. A set A is an **ancestral confounded set** for Y if $A = \mathrm{An}_{G[A]}(Y) = C_{G[A]}(Y)$. Thus, within its induced subgraph, every member of A is both an ancestor of Y and is connected to Y via a bidirected path. Every node Y has a unique maximum ancestral confounded set, denoted $\mathrm{MACS}_G(Y)$, which contains every ancestral confounded set for Y (Shpitser & Pearl, 2008b).

A causal query is *identifiable* if it can be uniquely expressed as a functional of the observational distribution in all SCMs compatible with the assumed graph. Formally, for disjoint variable sets $X, Y \subseteq V$, the causal effect $P(Y \mid do(X = x))$ is identifiable from $P(V)$ in graph G if every pair of SCMs inducing the same observational distribution $P(V)$ and graph G also induce the same interventional distribution $P(Y \mid do(X = x))$. Shpitser & Pearl (2008a) provide a sound and complete algorithm for identifying interventional distributions in ADMGs. We consider distribution shifts represented by selection variables. Let S denote the shift variables and let Y denote the target variable. Following the graph surgery estimator framework, we use conditional interventional distributions as invariant predictors (Subbaswamy et al., 2019; Lee et al., 2023).

Definition 2.1 (Graph surgery estimator)**.** Let S be the shift variables and Y be the target variable. For any subsets $Q, W \subseteq V$, if $(Y \perp\!\!\!\perp S \mid W)_{G_{\overline{Q}}}$ and $P(Y \mid do(Q), W)$ is identifiable in G and $P(Y \mid do(Q), W) \neq P(Y)$, then $P(Y \mid do(Q), W)$ is called a **graph surgery estimator**.

Intuition. The intervention set Q removes unstable incoming mechanisms into the intervened variables, while the conditioning set W provides predictive information that remains invariant across environments. The identifiability condition guarantees that the resulting interventional distribution can be computed from observational data.

2.2 Object-Centric Learning and Causal-JEPA

Object-centric learning represents a scene as a set of object-level latent representations. A prominent class of methods, including Slot Attention (Locatello et al., 2020), produces such representations as unordered object slots through competitive attention. Since slot representations are permutation-equivariant with respect to slot ordering, a fixed slot index does not necessarily correspond to the same object across videos. This motivates the graph-to-representation alignment step used in our method. Causal-JEPA (C-JEPA) uses object-centric representations for latent world modeling (Nam et al., 2026). Given a history of frames, a frozen object-centric encoder extracts object slots, after which C-JEPA masks selected object-level latent trajectories while preserving an identity anchor for each masked object. The predictor is trained to reconstruct the masked history and future latent tokens. Let Z_t denote the set of object slots and optional auxiliary variables at time t, and let $\bar{Z}_{\mathcal{T}}$ denote the masked input sequence over the history-future interval $\mathcal{T}$. The predictor outputs $\hat{Z}_{\mathcal{T}} = f_\theta(\bar{Z}_{\mathcal{T}})$. The masked latent prediction objective is

$$\mathcal{L}_{\text{C-JEPA}} = \mathbb{E}\left[\sum_{\tau \in \mathcal{T}} \sum_{i=1}^{N} \mathbf{1}[\bar{z}_\tau^i \neq z_\tau^i] \left\| \hat{z}_\tau^i - z_\tau^i \right\|_2^2\right]. \tag{1}$$

This objective encourages the predictor to infer masked object states from surrounding object states and auxiliary variables. C-JEPA characterizes this predictive structure through the influence neighborhood. For a masked object state z_t^i, let $Z_{\mathcal{T}}^{(-i)}$ denote the observable tokens excluding the masked trajectory of object i, while retaining the identity anchor. The influence neighborhood $N_t(i)$ is a minimal subset of observable context tokens such that $p(z_t^i | Z_{\mathcal{T}}^{(-i)}) = p(z_t^i | N_t(i))$. Thus, the influence neighborhood captures the local set of visible tokens that is predictively sufficient for masked completion. However, predictive sufficiency does not imply causal invariance. A variable may belong to $N_t(i)$ because it is useful for masked prediction, even if conditioning on it creates non-invariant dependence under changes in slot-generating mechanisms. Therefore, C-JEPA's influence neighborhood does not by itself guarantee an invariant causal predictor. Our goal is to test and improve this

invariance by estimating graph-surgery predictors over learned object representations. Additional details on influence neighborhoods and their interpretation are in Appendix B.

2.3 MOTIVATING EXAMPLE

We motivate Do-CJEPA using an object-centric video prediction problem under an unknown mechanism shift. Consider a Pong-like dynamical system with three visible foreground objects: a ball (B), a left paddle (L), and a right paddle (R) (Bellemare et al., 2013). The underlying data-generating process contains low-level causal factors such as ball position, ball velocity direction, paddle positions, and score. Figure 1a illustrates the resulting factor-level causal graph over three timestamps. Object-centric representation learners do not directly observe these causal factors. Instead, they encode pixels into object-level representations. Let Z_t^B, Z_t^L, and Z_t^R denote the matched object slots corresponding to the ball, left paddle, and right paddle at time t. Figure 1(b) shows the induced slot-level causal graph. During this abstraction, low-level factors such as the previous ball velocity direction become unobserved and their effects are absorbed into the slot dynamics, inducing latent confounding represented by bidirected edges.

We consider a distribution shift represented by a selection variable S. Suppose two environments are observed, where $S = 0$ denotes the source environment and $S = 1$ denotes the target environment. At the factor level, the mechanism generating the right paddle may change across environments. At the slot level, this corresponds to the slot Z_t^R being a child of the selection variable S. The selection edge indicates which conditional mechanism may vary across environments. Following C-JEPA, an object trajectory is randomly selected and masked. Let Z_τ^r denote a masked target slot belonging to object $r \in \{B, L, R\}$. C-JEPA learns a conditional predictor of the masked object using its influence neighborhood. However, when the conditioning set contains variables whose mechanisms change across environments, the resulting predictor may exploit environment-specific correlations rather than stable object dynamics. C-JEPA does not guarantee that Z_t^R will never be included in its influence neighborhood based on training data from the source environment. Specifically, S is conditionally dependent with Z_t^L given Z_t^R under the assumption of faithfulness [1]. Thus, C-JEPA may not always learn an optimal predictor of Z_t^L. Fortunately, interventions allow us to *cut off* the effects of S by intervening on the child of S [2]. (Subbaswamy et al., 2019) has shown that we can perform invariant prediction on Z_t^L by computing interventional query from observed data e.g., $p(Z_t^L | do(Z_t^R), Z_{t+1}^R)$ when causal knowledge is available. Based on this insight, we introduce Do-CJEPA, which replaces conditional prediction targets with graph-surgery estimator constructed based on the slot-level causal graph.

3 DO-CJEPA: GRAPH-SURGERY-REGULARIZED OBJECT-CENTRIC WORLD MODEL

In this section, we will discuss the core ideas of Do-CJEPA. We also provide Figure 1 as an overview of the algorithm. The goal is to leverage causal knowledge that can be learned through causal discovery or domain knowledge about the underlying generating factors of the images in a video.

Graph-to-representation alignment. An object-centric encoder g_ψ maps frame X_t to slots $Z_t = \{Z_t^1, \ldots, Z_t^N\}$. We align persistent slot roles to the variables of an ADMG G. This study considers two types of causal structure: the fully known object-level ADMG and a partial ancestral graph that represents imperfect knowledge of the causal relations [3]. We do not assume that the knowledge of the interventional targets is given.

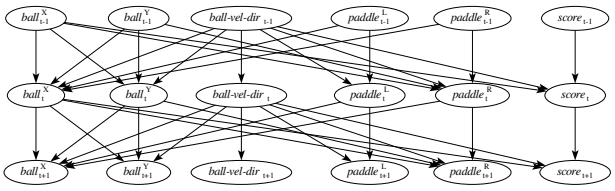
Graph-surgery query. For a target Y, let $M_Y = \mathrm{MACS}_G(Y)$ be its maximal ancestral confounded set and define

$$I_Y = \mathrm{pa}_G(M_Y) \setminus M_Y. \tag{2}$$

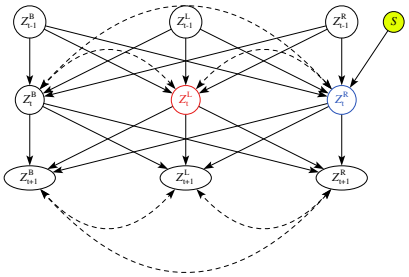
[1] Faithfulness posits that if A and B are m-connecting given C in the graph G, they are conditionally dependent given C. For details, please see Jaber et al. (2020).

[2] Although conditioning on Z_{t-1}^R will be sufficient to m-separate S from Z_t^L in this example. We provide an example where conditioning alone is not sufficient in general in the Appendix D

[3] More details on PAGs are provided in Appendix E



(a) Original causal relations in the data-generating process for the first three timestamps $t-1$ (top) through t (bottom).



(b) Slot-level causal graph for the first three timestamps. The ball velocity direction (`ball-vel-dir`) from the previous timestamp becomes a hidden confounder. S is a selection variable to denote a distribution shift on the conditional of Z_t^R.

Figure 1: **Left**: the original factor-level causal relation among simulator variables (Bellemare et al., 2013). **Right**: the induced slot-level causal graph used by Do-CJEPA. The slot graph abstracts the low-level causal factors into object slots Z^B, Z^L, Z^R, while a latent variable accounts for unobserved temporal information such as the previous ball velocity direction. The red node is the target token and the blue node is a shifted variable.

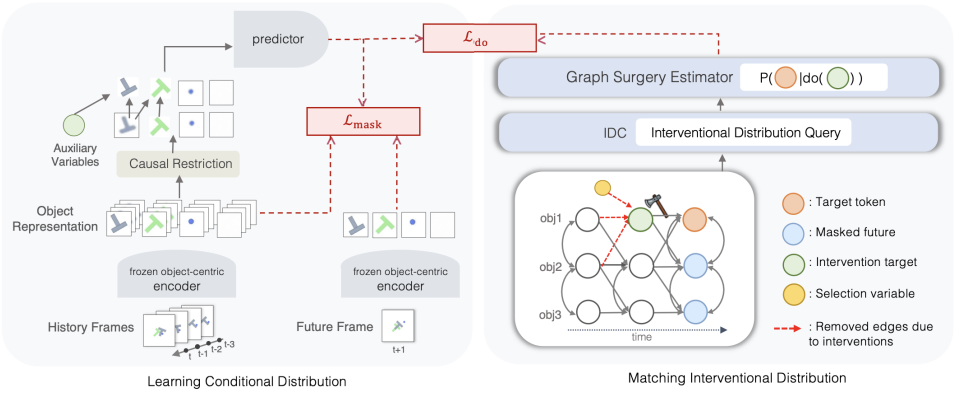


Figure 2: Overview of Do-CJEPA. The model estimates interventional distributions from observed data through graph surgery with a slot-level causal graph and uses that to compute a loss term to regularize C-JEPA objective. It also restricts the predictor set according to the causal structure in the original C-JEPA objective function.

Do-JEPA applies ID to $P(Y \mid do(I_Y))$ and uses an ID-GEN estimator trained only based on data from the source environment to estimate its mean $\widehat{\mu}_Y(I_Y)$ (Shpitser & Pearl, 2008a; Rahman et al., 2024). This construction is target-specific and does not require identifying which mechanism will change at test time given the ADMG. When the query is not identifiable, we will use $P(Y|I_Y)$ instead. Such cases only occur when there is an unobserved confounder between the outcome and one of the predictors at the previous time step. According to the Theorem 6 in (Shpitser & Pearl, 2008b), S and Y can only be $m^\star$-separated if and only if $p(Y|do(I_Y))$ is identifiable in G. Here, $m^\star$-separation denotes that Y and S are m-separated in G where the incoming edges of I_Y are removed.

Causal restriction. C-JEPA can use every unmasked history slot to complete a target. In contrast, we restrict the predictors to only the slots containing its parents at the immediately preceding time frame. We enforce this at every transformer layer. When an ADMG is not available, we then restrict the input to be possible parents at the preceding time frame from a partial ancestral graph at the slot-level [4].

[4] A possible parent set of Y is the set of adjacent nodes S such that Y does not have the following edges with X: $Y \to X$, $Y \leftrightarrow X$ for every $X \in S$ in the PAG.

Absolute interventional target. Let h_Y map a predicted slot into the coordinate in which the graph and numerical estimator are defined. For the known object graph, h_Y is the identity on the target slot. When an ADMG is not available, we use a causal representation learning algorithm named G-CaRL (Morioka & Hyvärinen, 2024) to supply anonymous factor coordinates and source standard deviation s_Y. We use the L-PCMCI causal discovery algorithm on the learned factors (Gerhardus & Runge, 2020) to obtain a partial ancestral graph. When a numerical interventional target is used, the model minimizes

$$\mathcal{L}_{\mathrm{do}} = \frac{1}{|\mathcal{T}_{\mathrm{ID}}|} \sum_{Y \in \mathcal{T}_{\mathrm{ID}}} \frac{\left\| h_Y(\widehat{Z}_t) - \mathrm{sg}[\widehat{\mu}_Y(I_Y)] \right\|_2^2}{d_Y s_Y^2}, \tag{3}$$

where d_Y is the target-coordinate dimension, $s_Y = 1$ for the known-slot estimator, and sg denotes stop-gradient. The realized-slot C-JEPA loss is retained, giving

$$\mathcal{L}_{\text{Do-CJEPA}} = \mathcal{L}_{\text{CJEPA}} + \lambda_{\mathrm{do}} \mathcal{L}_{\mathrm{do}}. \tag{4}$$

Optional rollout-matched training. The rollout version adds a source-only four-step objective. From a sampled source boundary b, it recursively feeds predictions back as context and compares the four generated slots with the observed source continuation:

$$\mathcal{L}_{\mathrm{roll}} = \frac{1}{4} \sum_{j=1}^{4} \left\| \widehat{Z}^{(j)}_{b+j-1} - Z_{b+j-1} \right\|_2^2, \qquad \mathcal{L}_{\mathrm{total}} = \mathcal{L}_{\text{Do-CJEPA}} + \lambda_{\mathrm{roll}} \mathcal{L}_{\mathrm{roll}}. \tag{5}$$

The same graph-restricted one-step predictor is used at every recursion. We compare Do-CJEPA with and without this term to separate one-step causal prediction from robustness to self-generated histories.

4 Experiments

In this experiment, we use the interventional Pong dataset (Lippe et al., 2022). The original causal graph is shown in Figure 1b. Specifically, we use 16-frame object-centric Pong trajectories containing a ball (B), left paddle (L), and right paddle (R). A one-time soft intervention changes the right-paddle state at a uniformly sampled timestamp $t^\star$. Every later state is generated by applying the ordinary transition mechanism to the modified state. Thus, the intervention is not repeatedly applied at every timestep, but its effect persists through the rollout. We train only on $10{,}000$ source trajectories, select checkpoints on $1{,}000$ source-validation trajectories, and evaluate on paired sets of $1{,}000$ source and $1{,}000$ shifted test trajectories. The models never receive $t^\star$, the intervened role, or the intervention value. Appendix C gives the data generation details. A SAVi encoder (Elsayed et al., 2022) trained only on source videos produces four 64-dimensional slots. Hungarian matching aligns the three foreground slots to B/L/R. All world models use these frozen slots and the same six-layer, 256-dimensional transformer, optimizer, 30-epoch schedule, and five seeds. We compare the following algorithms in the experiment.

- **C-JEPA (fixed) (Nam et al., 2026):** The baseline always all frames before the prediction frame. It retains C-JEPA's object-trajectory masking: a uniformly sampled subset of zero, one, or two foreground roles is hidden over history except for its initial identity anchor.
- **Do-JEPA (ADMG):**The known graph gives every target the three lag-one parent slots in Eq. 6. Each source batch samples a boundary $t \in \{2, \ldots, 15\}$, observes only frame $t-1$ through the graph-restricted information path, and predicts frame t. A source-trained eight-component diagonal Gaussian-mixture ID-GEN model provides the absolute target in Eq. 3, with $\lambda_{\mathrm{do}} = 1$. In Interventional Pong, $M_Y = \{Y\}$ and there are no same-time directed effects, so

$$P(Y_t \mid do(I_Y)) = P(Y_t \mid I_Y), \qquad I_Y = \{Z^B_{t-1}, Z^L_{t-1}, Z^R_{t-1}\}. \tag{6}$$

- **Do-CJEPA (PAG)** It uses G-CaRL and LPCMCI to obtain a PAG over learned latent factors of the images and proceeds with causal restriction by only using possible parents. The edges selected by at least three runs define the target-specific attention restriction. Appendix F gives the selected parents and edge frequencies.

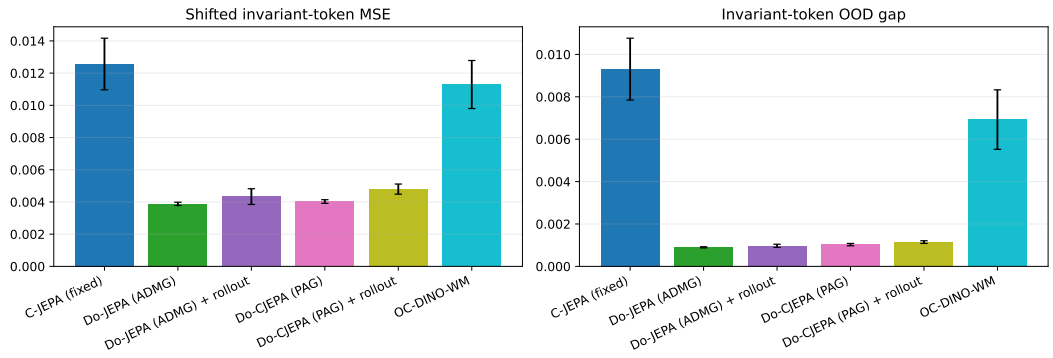


Figure 3: Final-step comparison at $t = 15$. Left: invariant-token MSE in the shifted environment. Right: shifted-minus-source OOD gap. Bars and error bars are means and sample SDs over five seeds; lower is better.

- **OC-DINO-WM:** It applies object-centric slot attention to frozen DINOv2 patch features, while retaining the original DINO-WM predictor and training objective to isolate the effect of object-centric representations. Following (Nam et al., 2026), we replace DINO-WM's patch tokens (Zhou et al., 2025) with the same frozen SAVi slots used by others.

Rollout versions. For both graph sources, we train one version with only the immediate-next-step objective and one version that additionally uses Eq. 5 with $\lambda_{\text{roll}} = 0.5$. The four-step term starts from a separately sampled valid source boundary and always predicts four consecutive timestamps; it does not repeat the same timestamp four times.

4.1 Final-Step Invariant Prediction

For environment $e \in \{0, 1\}$, we report invariant-token latent MSE at $t = 15$,

$$\text{MSE}_e^{\text{inv}} = \mathbb{E}_{\text{env}=e}\left[|\mathcal{T}_{\text{inv}}|^{-1} \sum_{Y \in \mathcal{T}_{\text{inv}}} \|\widehat{Y} - Y\|_2^2\right],$$

and $\Delta_{\text{OOD}} = \text{MSE}_1^{\text{inv}} - \text{MSE}_0^{\text{inv}}$. Evaluation annotations exclude the directly intervened right-paddle token only when $t^\star = 15$; they are not used in training. Figure 3 reports mean $\pm$ sample SD across five seeds.

Result. At the final prediction step, both graph-restricted models are substantially more accurate under the right-paddle intervention than fixed C-JEPA. Do-JEPA (ADMG) performs best, while Do-CJEPA (PAG) remains within 3.8% of its shifted invariant MSE and reduces the error of fixed C-JEPA by 67.9%. Their errors also increase much less between the source and shifted test sets. This improvement is consistent with the intended mechanism: target-specific attention prevents invariant predictions from depending on the intervened right-paddle slot, whereas fixed C-JEPA can exploit source-only cross-object correlations that do not survive the intervention.

4.2 Intervention-Aligned One-Step Prediction

At every evaluation timestamp t, each model receives the observed prefix $Z_{0:t-1}$ and predicts only Z_t. We never feed back the predictions. We align examples by $k = t - t^\star$. For $k < 0$, frame t precedes the intervention and is unaffected. At $k = 0$, the right paddle is directly intervened and is excluded only from the invariant-token curve. For $k > 0$, the observed prefix contains the realized consequences of the earlier intervention. The fixed C-JEPA checkpoint remains the model trained on $0{:}14 \to 15$; it is not retrained at each evaluation timestamp.

Result. Across the five post-intervention steps, Do-JEPA (ADMG) reduces invariant prediction error by 90.1% relative to fixed C-JEPA, while Do-CJEPA (PAG) performs nearly identically. Restricting attention to possible causal parents prevents the intervened right-paddle slot from contam-

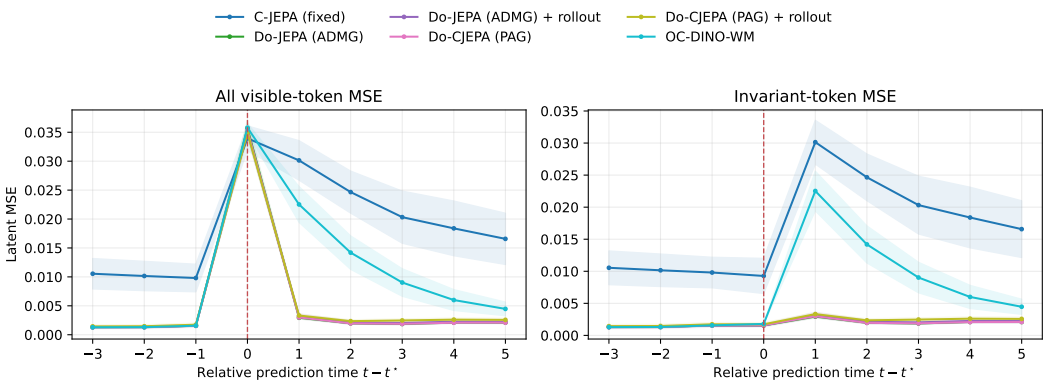


Figure 4: Teacher-forced next-step prediction aligned by $k = t - t^\star$. Left: all-token MSE. Right: invariant-token MSE, excluding the directly intervened token at $k = 0$. Curves and bands show five-seed mean $\pm$ sample SD. The reported post-shift metric averages invariant MSE over $k = 1, \ldots, 5$.

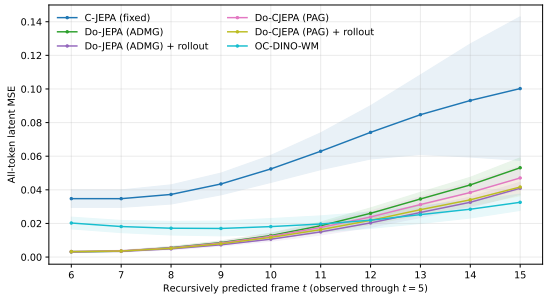


Figure 5: Post-intervention open-loop all-token latent MSE over the 62 shifted test trajectories with $t^\star = 5$. Models observe frames 0–5, including the intervened frame, and recursively consume their predictions for frames 6–15. Lines and bands show five-seed mean $\pm$ sample SD; $t = 15$ is the rightmost point.

inating invariant predictions. Rollout training is slightly worse under this teacher-forced evaluation because it trades one-step accuracy for robustness to recursively generated inputs.

4.3 Open-Loop Rollout

We isolate post-intervention propagation by selecting the 62 shifted test trajectories with $t^\star = 5$. All models observe $Z_{0:5}$, including the realized intervened frame, and recursively predict frames 6:15 without being told which variable changed or by how much. Figure 5 reports MSE at each generated timestamp; terminal $t = 15$ error follows ten recursive predictions. Figure 6 shows the prespecified seed-0, shifted-sequence-rank-16 example. SAVi decoder masks are rendered with fixed role colors; the qualitative row is illustrative, while the aggregate curve is the evidence.

Result. Based on Figure 5, at $t = 15$, graph-restricted prediction reduces open-loop error by roughly half relative to fixed C-JEPA. Four-step rollout training improves both variants further, leaving Do-JEPA (ADMG) and Do-CJEPA (PAG) about 59% below the baseline. This indicates that causal parent restrictions improve propagation of the observed intervention, while rollout exposure limits error accumulation from recursively predicted inputs. Figure 6 also shows the C-JEPA prediction on the ball position is affected by the unknown intervention at $t = 6$, whereas Do-CJEPA provides a more invariant prediction by having the ball position similar to the ground truth.

We provide an ablation study in Appendix H to understand the contributions of each part of Do-CJEPA. We have found that masking according to the causal structure is more significant to better prediction. This shows the benefits of leveraging causal knowledge of the underlying mechanism of the observed data. We additionally provide another experiment in Appendix I to showcase the

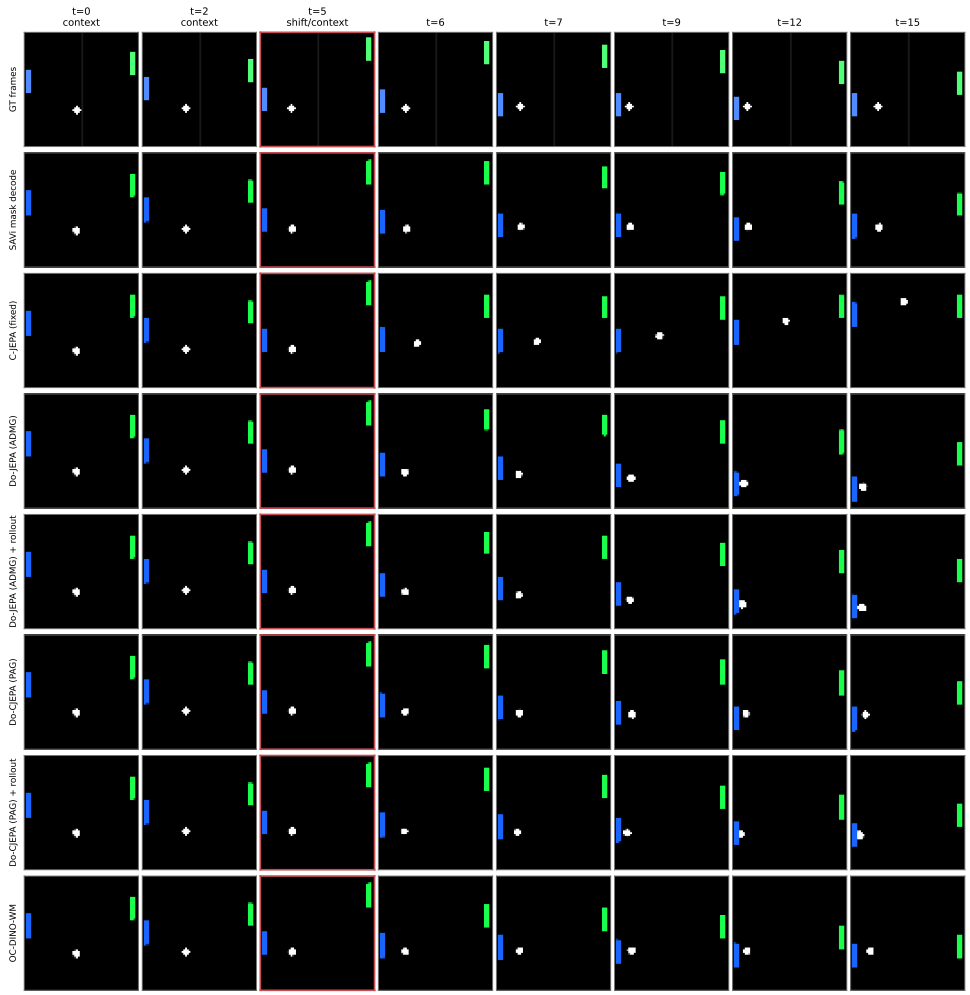


Figure 6: Qualitative open-loop rollout for seed 0 and shifted sequence rank 16. Columns show $t = 0, 2, 5, 6, 7, 9, 12, 15$. Frames 0–5 are observed and frames 6–15 are generated recursively; the red border marks the observed intervention at $t = 5$. Rows compare raw ground truth, SAVi mask rendering, fixed C-JEPA, Do-JEPA (ADMG), and Do-CJEPA (PAG), with and without rollout training; OC-DINO-WM on the same SAVi slots is shown in the final row.

utility of Do-CJEPA when the causal relations are not known. We show that Do-CJEPA provides more robust predictions on variables that are not directly affected by mechanism shifts.

5 Conclusions

We have introduced Do-CJEPA, which incorporates graph surgery estimation into an object-centric representation learning objective by restricting the masking information according to the causal structure without knowing the interventional targets. Do-CJEPA substantially outperforms C-JEPA both quantitatively and qualitatively based on the Interventional Pong benchmark and the additional PHYRE experiment. Do-CJEPA has smaller OOD gap between training and test errors. We additionally provide a version of Do-CJEPA when causal knowledge is not readily available or only partial knowledge of the underlying system is known via causal discovery. The limitation of the work to world modeling is its reliance on object-centric learning algorithms and causal discovery algorithms when the causal structure is unknown. Matching the low-dimensional variables with the slots may also impose additional limitations due to causal representation learning.

AI USE STATEMENT

In this work, we used generative AI tools for conducting the experiments and proofreading the paper. We have not used generative AI tools for generating the original idea of the paper. We have reviewed all AI-assisted work. We checked LLM-generated research ideas for potential plagiarism through a manual literature survey. All LLM-generated code was verified and tested for correctness. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The code for the experiments is provided, and the steps to reproduce them are also included in the README markdown file.

REFERENCES

Anton Bakhtin, Laurens van der Maaten, Justin Johnson, Laura Gustafson, and Ross Girshick. PHYRE: A new benchmark for physical reasoning. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of artificial intelligence research*, 47: 253–279, 2013.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9650–9660, 2021.

Gamaleldin Elsayed, Aravindh Mahendran, Sjoerd Van Steenkiste, Klaus Greff, Michael C Mozer, and Thomas Kipf. Savi++: Towards end-to-end object-centric learning from real-world videos. *Advances in Neural Information Processing Systems*, 35:28940–28954, 2022.

Fan Feng, Phillip Lippe, and Sara Magliacane. Learning interactive world model for object-centric reinforcement learning. *Advances in Neural Information Processing Systems*, 38:89827–89862, 2026.

Stefano Ferraro, Pietro Mazzaglia, Tim Verbelen, and Bart Dhoedt. Focus: object-centric world models for robotic manipulation. *Frontiers in Neurorobotics*, 19:1585386, 2025.

Juan L Gamella and Christina Heinze-Deml. Active invariant causal prediction: Experiment selection through stability. *Advances in Neural Information Processing Systems*, 33:15464–15475, 2020.

Andreas Gerhardus and Jakob Runge. High-recall causal discovery for autocorrelated time series with latent confounders. *Advances in neural information processing systems*, 33:12615–12625, 2020.

David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL `https://proceedings.neurips.cc/paper_files/paper/2018/file/2de5d16682c3c35007e4e92982f1a2ba-Paper.pdf`.

Amin Jaber, Murat Kocaoglu, Karthikeyan Shanmugam, and Elias Bareinboim. Causal discovery from soft interventions with unknown targets: Characterization and learning. *Advances in neural information processing systems*, 33:9551–9561, 2020.

Thomas Kipf, Elise Van der Pol, and Max Welling. Contrastive learning of structured world models. *arXiv preprint arXiv:1911.12247*, 2019.

Thomas Kipf, Gamaleldin F Elsayed, Aravindh Mahendran, Austin Stone, Sara Sabour, Georg Heigold, Rico Jonschkowski, Alexey Dosovitskiy, and Klaus Greff. Conditional object-centric learning from video. *arXiv preprint arXiv:2111.12594*, 2021.

Murat Kocaoglu, Christopher Snyder, Alexandros G. Dimakis, and Sriram Vishwanath. CausalGAN: Learning causal implicit generative models with adversarial training. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=BJE-4xW0W.

Kenneth Lee, Md Musfiqur Rahman, and Murat Kocaoglu. Finding invariant predictors efficiently via causal structure. In *Uncertainty in Artificial Intelligence*, pp. 1196–1206. PMLR, 2023.

Anson Lei, Bernhard Schölkopf, and Ingmar Posner. Spartan: A sparse transformer world model attending to what matters. *Advances in Neural Information Processing Systems*, 38:154089–154114, 2026.

Phillip Lippe, Sara Magliacane, Sindy Löwe, Yuki M Asano, Taco Cohen, and Stratis Gavves. CITRIS: Causal identifiability from temporal intervened sequences. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 13557–13603. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/lippe22a.html.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *Advances in neural information processing systems*, 33:11525–11538, 2020.

Hiroshi Morioka and Aapo Hyvärinen. Causal representation learning made identifiable by grouping of observational variables. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 36249–36293. PMLR, 2024.

Malte Mosbach, Jan Niklas Ewertz, Angel Villar-Corrales, and Sven Behnke. Sold: Slot object-centric latent dynamics models for relational manipulation learning from pixels. *arXiv preprint arXiv:2410.08822*, 2024.

Heejeong Nam, Quentin Le Lidec, Lucas Maes, Yann LeCun, and Randall Balestriero. Causal-jepa: Learning world models through object-level latent masking, 2026. URL https://arxiv.org/abs/2602.11389.

Yosuke Nishimoto and Takashi Matsubara. Object-centric world models for causality-aware reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 24585–24593, 2026.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Judea Pearl. *Causality*. Cambridge university press, 2009.

Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 78(5):947–1012, 2016.

Md Musfiqur Rahman, Matt Jordan, and Murat Kocaoglu. Conditional generative models are sufficient to sample from any causal effect estimand. *Advances in Neural Information Processing Systems*, 37:77269–77315, 2024.

Ilya Shpitser and Judea Pearl. Identification of joint interventional distributions in recursive semi-markovian causal models. In *AAAI*, pp. 1219–1226, 2006.

Ilya Shpitser and Judea Pearl. Complete identification methods for the causal hierarchy. 2008a.

Ilya Shpitser and Judea Pearl. Dormant independence. In *23rd AAAI Conference on Artificial Intelligence and the 20th Innovative Applications of Artificial Intelligence Conference, AAAI-08/IAAI-08*, pp. 1081–1087, 2008b.

Adarsh Subbaswamy, Peter Schulam, and Suchi Saria. Preventing failures due to dataset shift: Learning predictive models that transport. In Kamalika Chaudhuri and Masashi Sugiyama (eds.), *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pp. 3118–3127. PMLR, 16–18 Apr 2019. URL https://proceedings.mlr.press/v89/subbaswamy19a.html.

Jin Tian and Judea Pearl. A general identification condition for causal effects. In *Aaai/iaai*, pp. 567–573, 2002.

Angel Villar-Corrales, Ismail Wahdan, and Sven Behnke. Object-centric video prediction via decoupling of object dynamics and interactions. In *2023 IEEE International conference on image processing (ICIP)*, pp. 570–574. IEEE, 2023.

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.

Andrii Zadaianchuk, Maximilian Seitzer, and Georg Martius. Object-centric learning for real-world videos by predicting temporal feature similarities. *Advances in neural information processing systems*, 36:61514–61545, 2023.

Jiji Zhang. Causal reasoning with ancestral graphs. *Journal of Machine Learning Research*, 9(7), 2008.

Artem Zholus, Yaroslav Ivchenkov, and Aleksandr Panov. Factorized world models for learning causal relationships. In *ICLR2022 Workshop on the Elements of Reasoning: Objects, Structure and Causality*.

Artem Zholus, Yaroslav Ivchenkov, and Aleksandr Panov. Factorized world models for learning causal relationships. In *ICLR2022 Workshop on the Elements of Reasoning: Objects, Structure and Causality*, 2022.

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.

A Related Works

A.1 Object-centric World Models for Dynamics Learning

Learning predictive world models for dynamic scenes requires representations that capture both individual entities and their interactions (Caron et al., 2021; Oquab et al., 2023). Existing approaches address this challenge in several ways. Some methods introduce weak supervision over entities (Zholus et al., 2022) or use reinforcement-learning-specific architectures (Ferraro et al., 2025), while others build more general self-supervised world models on top of object-centric representations. A common strategy is to explicitly separate object self-dynamics from object-object interactions through architectural factorization (Mosbach et al., 2024; Feng et al., 2026). Other approaches reduce reliance on reconstruction losses and instead learn compact latent dynamics directly. For example, SlotFormer (Wu et al., 2022) predicts future object slots without pixel reconstruction, but does not explicitly constrain the learned interaction structure. C-SWM (Kipf et al., 2019) incorporates relational structure, but relies on a fixed graph that may limit flexibility. More recently, SPARTAN (Lei et al., 2026) learns latent-space world models while encouraging selective interactions through sparse attention. These works collectively show the importance of modeling interactions in object-centric dynamics, but they do not directly enforce interaction-aware prediction through the learning objective itself. Nam et al. (2026) proposes a flexible object-centric world model that extends masked joint embedding prediction from image patches to objective-centric representations. Object-level latent masking allows C-JEPA to capture interaction-dependent dynamics.

A.2 Invariant Causal Prediction

A major challenge in machine learning is to deploy an algorithm that generalizes well to unseen data. Peters et al. (2016) proposes invariant causal prediction based on the idea that the conditional distribution of a target Y given its direct causal predictors should remain stable across environments. It searches over predictor subsets and keeps those whose predictive relationship with Y is statistically invariant across environments, yielding conservative confidence statements for causal predictors without requiring the intervention targets or a full causal graph to be known. Subbaswamy et al. (2019) introduced graph surgery estimators, which construct stable predictors by using hypothetical interventions to remove dependence on mechanisms that may vary under distribution shift. These interventions are not physically carried out. Instead, their corresponding interventional distributions are recovered from observational data using causal identification (ID) algorithms (Tian & Pearl, 2002; Shpitser & Pearl, 2006). However, the proposed algorithm performs a subset search and each time requires using ID algorithm, which is computationally expensive. Lee et al. (2023) alleviates this issue by proposing an algorithm with incorporating greedy search methods given a causal structure.

B Influence Neighborhoods in C-JEPA

C-JEPA formalizes masked latent prediction through the notion of an influence neighborhood (Nam et al., 2026). For a masked object state z_t^i, let $Z_{\mathcal{T}}^{(-i)}$ denote the observable tokens in the history-future window excluding the masked trajectory of object i, while retaining the identity anchor. The influence neighborhood $N_t(i)$ is defined as a minimal sufficient subset of observable context tokens satisfying

$$p(z_t^i \mid Z_{\mathcal{T}}^{(-i)}) = p(z_t^i \mid N_t(i)).$$

Minimality means that no strict subset of $N_t(i)$ satisfies the same conditional sufficiency relation. In other words, removing any variable from $N_t(i)$ changes the conditional distribution of the masked target.

Under the squared-error masked prediction objective, the Bayes-optimal predictor is given by the conditional mean

$$\hat{z}_t^{i,*} = \mathbb{E}\left[z_t^i \mid Z_{\mathcal{T}}^{(-i)}\right].$$

Using the defining property of the influence neighborhood,

$$\hat{z}_t^{i,*} = \mathbb{E}\left[z_t^i \mid N_t(i)\right].$$

Therefore, the influence neighborhood can be interpreted as the minimal set of visible tokens required to achieve Bayes-optimal masked prediction.

Importantly, the influence neighborhood is a predictive concept rather than a causal one. Membership in $N_t(i)$ is determined by predictive utility for reconstructing masked tokens and does not necessarily correspond to a stable causal predictor. Variables may enter the influence neighborhood because they provide statistical information about the target, even when such information is mediated through latent confounding or environment-dependent mechanisms. Consequently, predictive sufficiency alone does not guarantee invariance under distribution shift.

C Dataset Generation

The environment contains three foreground objects: a ball (B), a left paddle (L), and a right paddle (R). The latent simulator state contains factors including ball position, ball velocity direction, left-paddle position, right-paddle position, and score variables. Ball velocity is not supplied as a separate slot or auxiliary input, although a sequence encoder may encode information about it from motion in the images. It is therefore omitted as an explicit node in the object-level graph rather than guaranteed to be absent from the learned ball slot.

We generate matched-initial-condition trajectory pairs. For each initial state, the source copy follows the original dynamics and the shifted copy receives a soft intervention to the right-paddle vertical coordinate at a shared timestamp $t^\star \sim \mathrm{Uniform}\{1, \ldots, 15\}$. The two copies consume different process-noise draws and hence are not exact same-noise counterfactual pairs. The train, validation, and test archives contain respectively 10,000, 1,000, and 1,000 initial-condition pairs, with one source and one shifted trajectory per pair. At the object level, the corresponding selection graph marks the mutable mechanism as

$$F \to Z_{t^\star}^R.$$

At the factor level, this changes the mechanism generating $\texttt{pright_y}_{t^\star}$. After the intervention, the simulator continues from the modified state so downstream frames reflect the shift. Source and shifted trajectories have labels $\mathrm{env} = 0$ and $\mathrm{env} = 1$, respectively. The world models and ID-GEN estimators train only on source trajectories; evaluation uses both environments. The current benchmark applies a one-time soft state intervention after the ordinary transition into $t^\star$:

$$U^\star = 2B - 1, \quad B \sim \mathrm{Beta}(2, 8), \qquad \texttt{pright_y}_{t^\star} \leftarrow \mathrm{clip}\Big(0.15\,\widetilde{\texttt{pright_y}}_{t^\star} + 0.85\,U^\star\Big),$$

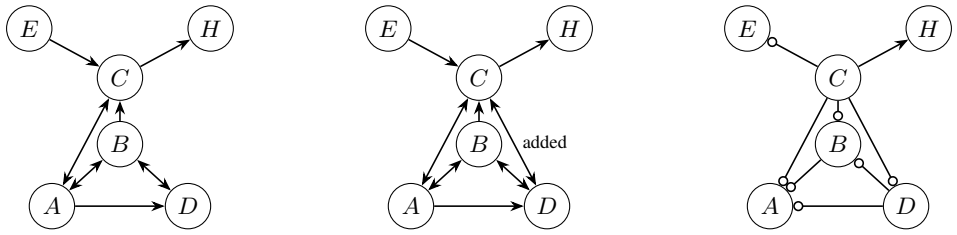
where $\widetilde{\texttt{pright_y}}_{t^\star}$ is the value after the ordinary transition and -1 is the top of the rendered screen. The simulator then propagates this perturbed state under the original transition rule. Thus, the implemented shift is a one-time state perturbation with downstream consequences rather than a persistent replacement of the transition mechanism. The mask $F \to Z_{t^\star}^R$ is stored for both members of a pair, while $F_{t^\star} = 1$ only in the shifted member. ID-GEN, the causal-discovery pipeline, the world models, and the revised SAVi representation learner use only source trajectories.

D An example where a graph surgery estimator is needed



(a) Original graph. (b) Graph with the incoming edge to X_t removed.

Figure 7: Temporal causal graphs before and after removing the incoming edge to X_t.



(a) A nonmaximal ADMG $\mathcal{G}$. (b) The corresponding MAG $\mathcal{M}$. (c) The PAG $\mathcal{P}$ representing $[\mathcal{M}]$.

Figure 8: Relationship among an ADMG, its maximal extension, and the PAG representing the MAG equivalence class. **Left:** C and D cannot be m-separated even though they are nonadjacent, so $\mathcal{G}$ is not maximal. **Middle:** adding $C \leftrightarrow D$ produces the MAG $\mathcal{M}$ without changing the encoded m-separation relations. **Right:** the PAG preserves only endpoint marks shared by every MAG in Markov equivalence class $[\mathcal{M}]$. For example, $C \rightarrow H$ is compelled, while $A \circ\!\!\rightarrow C$ fixes the arrowhead at C but leaves the endpoint at A unresolved. The edges among A, B, and D have unresolved endpoints at both ends.

Consider the example shown by Figure 7. If the outcome variable is X_{t+1}, no subset of observed variables can m-separate X_{t+1} from S in the presence of latent confounders. Here, we can predict the outcome X_{t+1} by estimating $p(X_{t+1}|do(X_t))$ via the adjustment formula (Pearl, 2009) e.g., $p(X_{t+1} \mid \mathrm{do}(X_t = x_t)) = \sum_{x_{t-1}} p(X_{t+1} \mid X_t = x_t,\, X_{t-1} = x_{t-1})\, p(X_{t-1} = x_{t-1})$ from observed data without the dependency of S. This setup is plausible when there are unobserved confounders at the previous time steps that can have a direct effect on both X_{t+1} and X_{t+1}.

E A PRIMER ON ADMGS, MAGS, AND PAGS

Acyclic directed mixed graphs (ADMGs), maximal ancestral graphs (MAGs), and partial ancestral graphs (PAGs) describe related but distinct levels of causal and statistical information. We briefly review their meanings because treating a PAG as though it were a fully specified ADMG can lead to incorrect identification claims (Zhang, 2008).

Acyclic directed mixed graphs. An *acyclic directed mixed graph* contains directed edges $X \rightarrow Y$ and bidirected edges $X \leftrightarrow Y$, with no directed cycle. When an ADMG is obtained by latent projection of an underlying causal DAG, a directed edge records a directed causal pathway among the observed variables, whereas a bidirected edge records latent common-cause structure. An ADMG need not be maximal, that is, two nonadjacent variables can remain dependent conditional on every subset of the remaining variables.

Maximal ancestral graphs. An ancestral graph additionally requires that an arrowhead at Y on an edge between X and Y imply that Y is not an ancestor of X. A *maximal ancestral graph* is an ancestral graph in which every nonadjacent pair of variables is m-separated by at least one conditioning set. Maximality does not mean that the graph is globally as dense as possible. It means that every missing edge corresponds to a conditional-independence implication.

Partial ancestral graphs. A *partial ancestral graph* represents an entire Markov equivalence class of MAGs: all MAGs represented by the PAG have the same m-separation relations among the observed variables. The endpoint marks have the following interpretation:

- a tail "$-$" is shared by every MAG in the equivalence class;
- an arrowhead "$>$" is shared by every MAG in the equivalence class; and
- a circle "$\circ$" denotes an endpoint whose mark differs across members of the equivalence class.

Thus, $X \rightarrow Y$ in a PAG is invariant across the represented MAGs, whereas $X \circ\!\!\rightarrow Y$ fixes the arrowhead at Y but leaves the endpoint at X unresolved. Depending on the equivalence class, $X \circ\!\!\rightarrow Y$ may correspond to $X \rightarrow Y$ or $X \leftrightarrow Y$. Likewise, $X \circ\!\!-\!\!\circ Y$ leaves both endpoints unresolved. A circle mark records orientation ambiguity under observational equivalence.

Implication for Do-CJEPA. The distinction is important for our two model variants. Do-JEPA (ADMG) assumes a particular ADMG, so parent sets, maximal ancestral confounded sets, and an ID functional can be defined on that graph. Do-CJEPA (PAG), in contrast, receives an equivalence-class representation containing unresolved circle endpoints. We therefore interpret its learned structure conservatively as a collection of possible-parent restrictions. The reported PAG variant uses those restrictions as an attention mask and sets $\lambda_{\text{do}} = 0$. It does not treat the learned PAG as a fully oriented ADMG or claim that the PAG alone identifies the same numerical interventional distribution.

F Implementation Detail

The implemented configuration uses $T = 16$, 64×64 RGB frames, four raw SAVi slots, and three matched foreground slots of dimension 64. Auxiliary CITRIS-style interventions are disabled in its original code. Source and shifted members share an initial condition but not an identical process-noise stream. The systematic intervention is the one-time 0.85-strength upper-biased perturbation of $\texttt{pright_y}_{t^\star}$ described above in Section C.

All comparisons use five random seeds. ID-GEN, G-CaRL, the conditional factor estimators, and every world model are trained only on source ($\text{env} = 0$) slots. Each world model is trained for 30 epochs. C-JEPA always observes $Z_{0:14}$ and predicts Z_{15}. Each graph-restricted batch instead samples an immediate-next-step boundary uniformly from $\{2, \ldots, 15\}$ and truncates the batch after that target. Do-JEPA (ADMG) and Do-CJEPA (PAG) use the same optimizer and six-layer predictor capacity. The shifted test split is never used for fitting, graph aggregation, hyperparameter selection, or checkpoint selection.

Do-JEPA (ADMG) uses $\lambda_{\text{do}} = 1$. Do-CJEPA (PAG) aggregates five source-only role-parent masks at threshold .5, uses the resulting hard layerwise parent restriction, and sets $\lambda_{\text{do}} = 0$. Non-rollout variants select the checkpoint with lowest ordinary source-validation prediction loss. Rollout variants add Eq. 5 with weight 0.5 and select checkpoints using source-validation one-step loss plus source open-loop MSE from two observed frames.

SAVi-style object-centric encoder Before training the C-JEPA and Do-CJEPA predictors, we train a lightweight SAVi-style object-centric encoder on the Interventional Pong videos **?**Elsayed et al. (2022). The encoder produces $K = 4$ raw slots per frame with slot dimension 64. We use a hidden dimension of 64 and train the encoder for 30 epochs with batch size 16. The model is initialized using first-frame ground-truth mask centers through the `gt_mask_centers` conditioning option. This conditioning is used only to obtain reliable object-aligned slots for the controlled comparison between C-JEPA and Do-CJEPA. The encoder is trained with a reconstruction loss together with supervised object-mask losses. Since the ball occupies very few pixels compared to the paddles and background, we use a stronger weight for the ball mask loss. The total SAVi encoder loss is $\mathcal{L}_{\text{SAVi}} = \mathcal{L}_{\text{recon}} + \lambda_{\text{mask}}\mathcal{L}_{\text{mask}} + \lambda_{\text{centroid}}\mathcal{L}_{\text{centroid}}$, where $\mathcal{L}_{\text{mask}}$ combines weighted binary cross-entropy and Dice loss over the three foreground object masks, and $\mathcal{L}_{\text{centroid}}$ penalizes the distance between predicted and ground-truth mask centroids. We set $\lambda_{\text{mask}} = 20.0, \lambda_{\text{centroid}} = 5.0$. The per-object mask weights are $w_{\text{ball}} = 30.0, w_{\text{left}} = w_{\text{right}} = 5.0$. After training, we extract raw SAVi slots for the train, validation, and test splits. Since raw slot indices are permutation-equivariant and do not necessarily correspond to semantic object identities, we match raw slots to object roles using Hungarian matching based on overlap between predicted slot masks and ground-truth object masks. The resulting matched slot tensors have shape $N \times T \times 3 \times 64$, where the third dimension corresponds to ball, left paddle, and right paddle. These matched object slots are then used as the input representation for both C-JEPA and Do-CJEPA.

The SAVi checkpoint used in the revised comparison is fitted only to the $10{,}000$ source trajectories. Its decoder-mask outputs, rather than its low-contrast RGB reconstruction, are rendered in the qualitative rollout figure with fixed Pong role colors.

The reported Do-JEPA (ADMG) experiment uses the conditional-generator base case of ID-GEN (Rahman et al., 2024). For $Y = Z_t^r$, the estimator receives the ordered parent slots $\{Z_{t-1}^B, Z_{t-1}^L, Z_{t-1}^R\}$ and a target-role embedding. It parameterizes an eight-component diagonal Gaussian mixture with hidden width 512, four residual MLP blocks, and no dropout, trained by negative log likelihood for 30 epochs with batch size 1024, Adam optimizer with learning rate 2×10^{-4},

and weight decay 10^{-5}. Its analytic mixture mean is cached for every source parent assignment and used as the stop-gradient target in Eq. 3.

For Do-CJEPA (PAG), each of five G-CaRL/LPCMCI runs supplies a factor-to-role possible-parent mask. The role-edge frequencies are $[(1, 1, .6), (.6, 1, .2), (1, .4, .8)]$ for ball, left-paddle, and right-paddle targets, respectively. Thresholding at .5 gives parent-index rows $[0, 1, 2]$, $[0, 1]$, and $[0, 2]$. This single source-derived mask is shared by the five world-model seeds, whose remaining initialization and minibatch randomness are independent. Consequently, the reported world-model standard deviation measures predictor-training variability conditional on the selected PAG structure; it does not estimate uncertainty over a fresh set of graph-discovery runs.

PHYRE mid-trajectory intervention implementation. The experiment in Appendix I cannot be implemented by serializing a PHYRE frame and restarting the public simulator: the public `Scene` omits linear and angular velocity. We therefore use PHYRE's pinned Box2D source and retain one world instance for the complete trajectory. The native simulation uses the original PHYRE constants: 60 solver steps per second, time step $1/60$, 15 velocity iterations, 20 position iterations, gravity -9.8, density .25, friction .5, restitution .2, angular damping .01, and zero linear damping. After the ordinary solver step corresponding to sampled frame $t = 5$, but before rasterization, we call Box2D's in-place body transform to add 16 simulator pixels to the dynamic green object's horizontal position. The transform leaves angle, linear velocity, and angular velocity unchanged, and the same world then advances normally. Rendering still uses PHYRE's own renderer and semantic color map.

We run factual and interventional copies for a fixed 15 sampled frames without the task-level early-stop check, while leaving the Box2D dynamics unchanged. The original PHYRE wrapper stops a successful trajectory after the task condition has held for 180 consecutive solver steps; using those variable lengths would change the test cohort at every rollout horizon. We retain each source early-stop length only as audit metadata and score the same 80 held-out trajectories at every reported frame. Runtime assertions verify that factual and interventional states are equal for frames $0{:}4$, that every native factual frame before the corresponding standard early-stop point matches the source archive exactly, and that the pre/post velocity vectors at $t = 5$ are equal. The generated audit metadata records the intervention body, factual and intervened positions, pre/post velocity, intervention frame, and paired action index. Frozen SAVi encoding provides a second check: all context slots are bit-identical before rollout. Because only the test generator changes, the already source-trained world-model checkpoints require no retraining.

All experiments were run on a local GPU server equipped with two NVIDIA GeForce RTX 3090 GPUs, each with 24GB of GPU memory.

G Algorithms

Let G be an ADMG over observed variables V, and let $P(V)$ denote the observational distribution. For $A \subseteq V$, $G[A]$ denotes the induced subgraph on A. We write G_X for the graph obtained by deleting incoming directed edges into X. For the IDC rule below, $G_{X,\underline{Z}}$ denotes the graph obtained from G_X by additionally deleting outgoing directed edges from Z. Let $\mathcal{D}(G)$ denote the set of the maximal set of variables connected by bidirected edges in G. We suppress value assignments when clear. For example, $\mathrm{ID}(Y, X, P, G)$ returns an expression for $P_x(y)$. Algorithm 3 is used by Do-CJEPA when a slot-level ADMG is available.

H Controlled Architecture and Component Ablations

All rows use source-only SAVi slots, random prediction boundaries $t \in \{2, \ldots, 15\}$, no masked history roles, the same 30-epoch schedule, source-validation checkpoint selection, and five matched seeds. "Mask" is the layerwise lag-one parent-attention restriction; "target" is the absolute ID-GEN mean loss. Capacity-matched OC-DINO-WM follows DINO-WM's frame-causal next-latent transition design (Zhou et al., 2025) after replacing patch tokens with the same frozen SAVi slots, as in the representation control of Nam et al. (2026). We match predictor capacity here rather than reproduce that paper's wider Push-T configuration. Terminal rollout observes frames $0{:}5$ for the 62 shifted trajectories with $t^\star = 5$ and recursively predicts $6{:}15$.

Algorithm 1 Training Do-JEPA with a Known ADMG

Require: Source video dataset $\mathcal{D}_{\mathrm{src}}$; frozen object-centric encoder E; known ADMG $\mathcal{G}$; predictor f_θ; interventional estimator $\widehat{\mu}$; loss weight λ_{do}

Ensure: Trained predictor parameters θ

1: Use graph surgery on $\mathcal{G}$ to determine the allowed lagged parent roles

$$\mathcal{P}_r = \mathrm{pa}_{\mathcal{G}}(Z_t^r) \cap \{Z_{t-1}^1, \ldots, Z_{t-1}^K\}$$

for every target role $r \in \{1, \ldots, K\}$

2: Compile $\{\mathcal{P}_r\}_{r=1}^K$ into a target-specific attention mask $M_{\mathcal{G}}$

3: **for** each training iteration **do**

4: Sample a source minibatch $\{X_{0:T}^{(n)}\}_{n=1}^B \sim \mathcal{D}_{\mathrm{src}}$

5: Encode the frames into frozen object slots:

$$Z_{0:T} \leftarrow \mathrm{sg}[E(X_{0:T})]$$

6: Sample a prediction boundary t and construct the observed history $Z_{<t}$

7: **for** each target role $r \in \{1, \ldots, K\}$ **do**

8: Select the graph-permitted inputs

$$I_t^r \leftarrow \left\{ Z_{t-1}^j : Z_{t-1}^j \in \mathcal{P}_r \right\}$$

9: Predict the target slot using graph-restricted attention:

$$\widehat{Z}_t^r \leftarrow f_\theta^r\left(Z_{<t}; M_{\mathcal{G}}\right)$$

10: Obtain the graph-surgery target:

$$\widetilde{\mu}_t^r \leftarrow \mathrm{sg}[\widehat{\mu}_r(I_t^r)]$$

11: Compute the realized-slot prediction loss:

$$\mathcal{L}_{\mathrm{JEPA}} = \frac{1}{BK} \sum_{n=1}^{B} \sum_{r=1}^{K} \frac{1}{d_r} \left\| \widehat{Z}_{t,n}^r - Z_{t,n}^r \right\|_2^2$$

12: Compute the graph-surgery loss:

$$\mathcal{L}_{\mathrm{do}} = \frac{1}{B|\mathcal{T}_{\mathrm{ID}}|} \sum_{n=1}^{B} \sum_{r \in \mathcal{T}_{\mathrm{ID}}} \frac{1}{d_r} \left\| \widehat{Z}_{t,n}^r - \widetilde{\mu}_{t,n}^r \right\|_2^2$$

13: Form the total objective:

$$\mathcal{L}_{\text{Do-JEPA}} = \mathcal{L}_{\mathrm{JEPA}} + \lambda_{\mathrm{do}} \mathcal{L}_{\mathrm{do}}$$

14: Update the predictor:

$$\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}_{\text{Do-JEPA}}$$

15: **Return** θ

Random-boundary training is a strong control: relative to fixed C-JEPA it reduces shifted invariant MSE by 36.0% and terminal rollout error by 71.5%. The same-slot OC-DINO-WM baseline is competitive in rollout but does not match Do-JEPA (ADMG)'s shifted one-step error. Within the matched factorial, however, the attention restriction accounts for essentially all of the full model's one-step gain; adding the absolute target does not improve over mask-only. The target-only model is close to matched C-JEPA. Moreover, generic four-step rollout training does not improve matched C-JEPA, while it improves full Do-JEPA (ADMG) from .05312 to .04092 but remains worse than matched C-JEPA at the terminal horizon. Omitting a true parent is damaging, particularly the intervened right-paddle parent. These results narrow the supported conclusion: graph-restricted information flow improves one-step robustness when the graph is correct; neither absolute interventional supervision nor open-loop superiority is established here.

Algorithm 2 $\mathrm{ID}(Y, X, P, G)$ Identification of unconditional causal effects (Shpitser & Pearl, 2006)

Input Disjoint sets $X, Y \subseteq V$, observational distribution P, ADMG G
Output Expression for $P(Y \mid \mathrm{do}(X))$ in terms of P, or FAIL

if $X = \emptyset$ **then**
  Return $\sum_{V \setminus Y} P(V)$
if $V \setminus \mathrm{An}_G(Y) \neq \emptyset$ **then**
  Return $\mathrm{ID}(Y, X \cap \mathrm{An}_G(Y), \sum_{V \setminus \mathrm{An}_G(Y)} P(V), G[\mathrm{An}_G(Y)])$
$W \leftarrow (V \setminus X) \setminus \mathrm{An}_{G_X}(Y)$
if $W \neq \emptyset$ **then**
  Return $\mathrm{ID}(Y, X \cup W, P, G)$
Let $\mathcal{D}(G[V \setminus X]) = \{D_1, \ldots, D_k\}$
if $k > 1$ **then**
  Return $\sum_{V \setminus (Y \cup X)} \prod_{i=1}^{k} \mathrm{ID}(D_i, V \setminus D_i, P, G)$
Let $\mathcal{D}(G[V \setminus X]) = \{D\}$
if $\mathcal{D}(G) = \{V\}$ **then**
  Return $\mathrm{FAIL}(G, G[D])$
if $D \in \mathcal{D}(G)$ **then**
  Let π be a topological ordering of G, and let $V_\pi^{(i-1)}$ be the variables preceding V_i in π
  Return

$$\sum_{D \setminus Y} \prod_{\{i : V_i \in D\}} P\left(v_i \mid v_\pi^{(i-1)}\right)$$

if there exists $D' \in \mathcal{D}(G)$ such that $D \subset D'$ **then**
  Let π be a topological ordering of G, and define

$$P'(D' \mid V \setminus D') = \prod_{\{i : V_i \in D'\}} P\left(v_i \mid v_\pi^{(i-1)} \cap D', v_\pi^{(i-1)} \setminus D'\right).$$

  Return $\mathrm{ID}(Y, X \cap D', P', G[D'])$

Algorithm 3 $\mathrm{IDC}(Y, X, Z, P, G)$ Identification of conditional causal effects (Shpitser & Pearl, 2008a)

Input Pairwise disjoint sets $X, Y, Z \subseteq V$, observational distribution P, ADMG G
Output Expression for $P(Y \mid \mathrm{do}(X), Z)$ in terms of P, or FAIL

if there exists $Z \in Z$ such that$(Y \perp\!\!\!\perp Z \mid X, Z \setminus \{Z\})_{G_{X,\underline{Z}}}$ **then**
  Return $\mathrm{IDC}(Y, X \cup \{Z\}, Z \setminus \{Z\}, P, G)$
else
  $P' \leftarrow \mathrm{ID}(Y \cup Z, X, P, G)$
  Return $\frac{P'}{\sum_Y P'}$

Target weight and correspondence. Holding the parent mask fixed, $\lambda_{\mathrm{do}} \in \{0, .1, .5, 1, 2\}$ gives shifted invariant MSE .00382, .00381, .00387, .00388, and .00393, respectively. Post-shift one-step MSE is likewise nearly flat (.00214, .00211, .00216, .00219, .00218), while terminal rollout MSE weakly worsens from .04586 at zero weight to .05312 at weight one. Thus the negative incremental result is not peculiar to $\lambda_{\mathrm{do}} = 1$; the small best mean at .1 is below one standard error of the zero-weight row and was not selected on shifted data. As a target-correspondence falsification, rotating ID-GEN means across examples at weight one raises shifted MSE to $.02300 \pm .00082$ and post-shift MSE to $.01906 \pm .00109$. An incorrect target is therefore actively harmful even though the correct target is redundant with the parent restriction in this benchmark.

Support and intervention stratification. Figure 9 bins the 1,000 shifted trajectories by (i) the standardized nearest-neighbor distance of their $t = 14$ parent frame to the 10,000 source parent frames and (ii) the paired right-slot change at $t^\star$. Each bin contains 250 trajectories. From the lowest to highest support-distance quartile, matched C-JEPA's all-token final-step MSE rises from .00618 to .01966, while Do-JEPA (ADMG) rises from .00580 to .00797. Across intervention-magnitude

Table 1: Controlled five-seed ablation (mean ± sample SD; lower is better). Inv. is final-step shifted invariant-token MSE; post-shift averages teacher-forced invariant MSE over $k = 1{:}5$.

Model	Shifted inv.	Post-shift	$t = 15$ rollout
C-JEPA, fixed boundary	.01256 ± .00160	.02202 ± .00409	.10023 ± .04271
C-JEPA, matched	.00804 ± .00114	.00724 ± .00246	**.02858 ± .00774**
OC-DINO-WM (SAVi; matched)	.01129 ± .00149	.01124 ± .00212	.03257 ± .00483
Mask only	**.00382 ± .00008**	**.00214 ± .00005**	.04586 ± .00565
Do target only	.00749 ± .00144	.00611 ± .00245	.02693 ± .01079
Full Do-JEPA (ADMG)	.00388 ± .00010	.00219 ± .00013	.05312 ± .00590
C-JEPA + four-step rollout	.01054 ± .00170	.01201 ± .00299	.03179 ± .00493
Do-JEPA (ADMG) + four-step rollout	.00433 ± .00049	.00237 ± .00033	.04092 ± .00358
Full, omit ball parent	.01453 ± .00021	.02986 ± .00144	.16225 ± .03353
Full, omit right parent	.02943 ± .00087	.03078 ± .00107	.08683 ± .01271

quartiles, the corresponding changes are .00450 → .02218 and .00331 → .01322. Both models degrade for large shifts, but Do-JEPA (ADMG) is less sensitive to departure from source support. These are descriptive diagnostics: nearest-neighbor distance in SAVi space is not a formal overlap test.

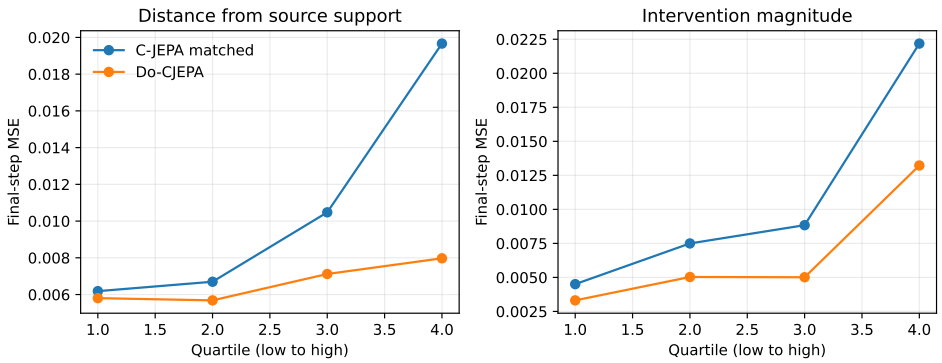


Figure 9: Final-step all-token MSE stratified by source-support distance (left) and intervention magnitude (right). Values average predictions over five seeds.

I PHYRE UNEXPECTED MID-TRAJECTORY OBJECT INTERVENTION

Here, we provide an additional controlled experiment for qualitative assessment where the causal graph of the generating factors of the images are completely unknown. We only take the first fold out of 24 folds in the dataset.

Using the original deterministic Box2D simulator in PHYRE v0.2.2 (Bakhtin et al., 2019), we take the first 25 tasks in the first fold and sample 20 successful ball actions per task from the public 100k-action cache, sample at one frame per 60 solver steps, vertically flip the raster as in public PHYRE preprocessing, and retain 15 128×128 frames. The first 20 tasks in the first fold supply 360 training and 40 validation trajectories. No intervened frame is used for representation learning, world-model training, PAG estimation, hyperparameter selection, or checkpoint selection.

For testing, the disjoint tasks from 21 to 24 supply 80 paired trajectories. The factual member uses the cached successful action. For its interventional member, we run the same task and red-ball action in one continuous Box2D world. Frames $t = 0{:}4$ evolve factually. At the sampling instant for $t = 5$, after the ordinary Box2D solver step and immediately before rendering, we apply

$$\mathrm{do}(x_5^{\mathrm{green}} = x_5^{\mathrm{green,factual}} + 16),$$

where coordinates are in the 256×256 simulator. We translate the existing body in place, preserve its angle, linear velocity, and angular velocity, and continue the same world under unchanged dynamics. The displayed shift is eight pixels at 128×128. This avoids the invalid alternative of reconstructing a new `Scene` at $t = 5$, because PHYRE serializes body position and orientation but not velocity.

The intervention is unexpected: every model receives only factual frames $t = 0{:}4$, receives neither the target nor the displacement, and recursively predicts $t = 5{:}10$. The realized $t = 5$ frame is not inserted into the open-loop history. Across all 100 generated test pairs, the factual and interventional RGB prefixes are pixel-identical through $t = 4$, the first changed render is always $t = 5$, and the pre/post intervention velocity vectors are bit-identical. The frozen SAVi slots are likewise exactly identical through $t = 4$. After excluding the first 20 tasks as described below, all 80 reported trajectories remain active at every evaluated frame $t = 5{:}10$. We obtain this fixed cohort by continuing both native Box2D worlds for all 15 sampled frames instead of invoking PHYRE's task-level success termination. The original early-stop length is retained as metadata but is not used as an evaluation mask, and no repeated padding frame is scored.

Validated object representation and rollout rendering. We use SlotFormer's released fold-0 PHYRE SAVi checkpoint (Wu et al., 2022), which has eight 128-dimensional slots and a spatial-broadcast RGB decoder that starts at 16×16 resolution. This is materially different from using a small Pong decoder on PHYRE: on the full-length mid-trajectory intervention set the released decoder has whole-frame RGB MSE $.00055$. On unmodified held-out videos, its decoded role centroids are within 0.44, 0.59, and 1.70 pixels of the green, red, and aggregated-blue simulator masks, respectively. Source-only decoder overlap identifies slot 1 as background, slots 6 and 4 as the red and green objects, and slots $\{0, 2, 3, 5, 7\}$ as components of the blue object. Every world model predicts the same seven foreground components. For visualization, we place them back at their original indices, hold only the black-background slot at its last observed value, and apply the frozen SAVi RGB decoder. The resulting model rows are decoded predictions, not simulator masks painted with role colors.

The released checkpoint was trained under PHYRE's fold 0 split. The first 20 tasks belong to its representation-training split and is therefore excluded from every reported test metric and figure. C-JEPA uses the public PHYRE choice of two masked slots. C-JEPA, Do-CJEPA (PAG), and OC-DINO-WM share the same frozen slots, six-layer 256-dimensional predictor capacity, optimizer, 100-epoch budget, source-validation checkpoint selection, five-frame context, and five seeds. OC-DINO-WM keeps the frame-causal next-slot objective while replacing patch tokens with these object-centric slots.

Learned PAG. We estimate role-level temporal PAGs from the frozen decoder-mask centroids (x, y) using five bootstrap LPCMCI runs with lags one and two and retain possible-parent edges appearing in at least half of the runs. The red-, green-, and blue-target parent frequencies are $(1, .2, .8)$, $(1, 1, 1)$, and $(1, 1, 1)$, respectively. Thresholding retains red and blue as possible parents of red,

and all three roles as possible parents of green and blue. These role relations are expanded to all SAVi components before constructing the layer-wise attention mask. Because PHYRE supplies no ground-truth temporal local graph, these learned dependencies must not be interpreted as causal-graph recovery.

Findings. Based on Figure 10, we see that Do-CJEPA (PAG) (5th row) managed to predict the red ball position successfully when we compare that with the ground truth under interventions (2nd row). We can see that C-JEPA has predicted the red ball more towards the right side, while OC-DINO-WM predicts the red ball more towards the left. The predictions on green ball are very inaccurate as expected as it is where the mechanism shifts get applied at $t = 5$.

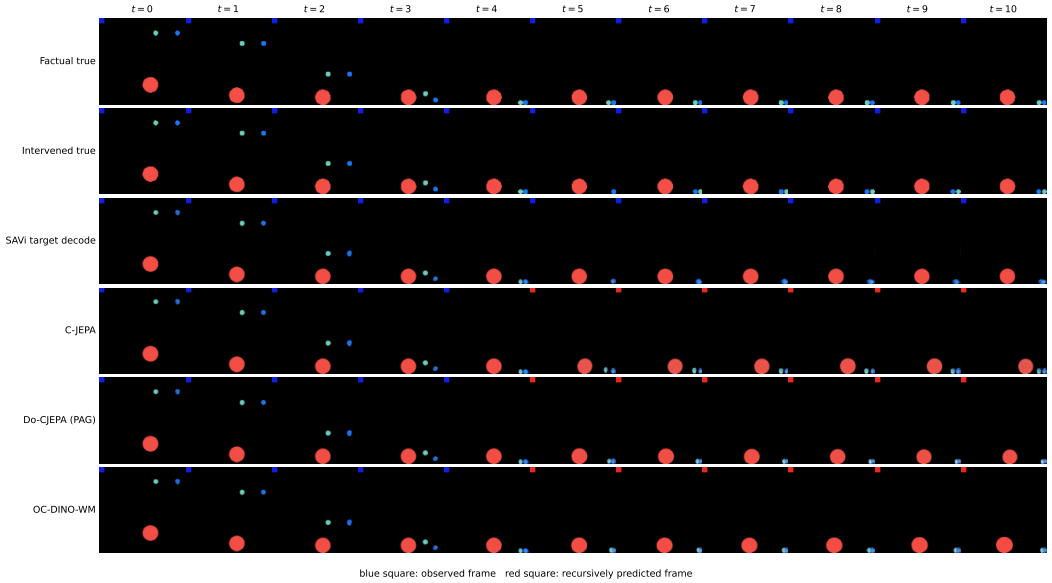


Figure 10: Paired PHYRE intervention and open-loop RGB rollout. Factual and interventional simulator rows are identical through $t = 4$; the green object is translated only at $t = 5$. The third row validates the frozen SAVi target decode. Model rows observe factual $t = 0{:}4$ (blue square), recursively predict $t = 5{:}10$ without seeing the displacement (red square), and use the same frozen SAVi RGB decoder. The seed-0 example is selected without model error as the pair with the largest ground-truth centroid response.