OCCAM: OBJECT-CENTRIC CAUSAL WORLD MODEL VIA VISION-LANGUAGE GRAPH GROUNDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Latent world models trained via joint-embedding predictive architecture (JEPA) excel at planning from pixels, but their transition functions are monolithic, lacking explicit representations of objects and interaction dynamics necessary for controllable and explainable embodied planning and reasoning. To address this issue, we introduce Object Centric CAusal World Model (**OCCAM**), a framework that integrates language-grounded causal structures directly into the transition function. **OCCAM** first uses open-vocabulary segmentation and tracking to obtain object-level latent states without hand-labeled masks. Then, **OCCAM** queries a vision-language model over motion-salient frames to build a sparse causal graph, validated by an independent conditional mutual information estimator. By combining language-grounded object abstraction with verified causal graph discovery, **OCCAM** enforces sparse, explainable state updates that isolate task-irrelevant entities, mitigating the extensive data requirements in classical causal discovery. We systematically evaluate **OCCAM**, analyzing dense, object-centric, and causal transition regimes across 12 tasks of increasing visual and relational complexity. Experimental results demonstrate that **OCCAM** achieves a highly interpretable, object-centric latent space while preserving goal-conditioned planning accuracy. Crucially, on the complex multi-object LIBERO benchmark, **OCCAM** achieves superior downstream task control compared to dense and object-centric baselines, indicating that the structured causal reasoning is essential for reliable control as scene complexity scales.

1 INTRODUCTION

World models provide a powerful abstraction for embodied intelligence, enabling agents to learn, predict, and reason over environmental dynamics by rolling out imagined trajectories in a learned representation rather than raw observation space (Ha & Schmidhuber, 2018; Hafner et al., 2020; Hansen et al., 2024). Latent world models built on the joint-embedding predictive architecture (JEPA) further advance this paradigm by avoiding pixel reconstruction entirely, predicting and planning directly within a compact latent space (LeCun, 2022; Assran et al., 2023; Bardes et al., 2024). However, the transition functions in these models typically operate as single monolithic networks that update the entire latent state without explicitly representing distinct entities or their interaction dynamics (Zhou et al., 2025; Maes et al., 2026). While recent advancements have explored structured representations, either by directly reconstructing pixels with object-centric models (Kipf et al., 2020; Lin et al., 2020; Veerapaneni et al., 2020), operating over hand-factored states (Wang et al., 2022; Zhu et al., 2022), or introducing structural priors into the latent space itself (Wu et al., 2023; Villar-Corrales et al., 2023; Nam et al., 2026; Feng et al., 2025; Nishimoto & Matsubara, 2026). Integrating explicit and interpretable causal structures into the transition dynamics of compact latent world models remains an ongoing challenge.

Even when causal structure is desired, discovering it from high-dimensional visual data is itself a bottleneck: classical causal discovery requires substantial data and clean, low-dimensional variables(Spirtes et al., 2001; Wang et al., 2022), and both break down once variables must first be extracted from raw pixels rather than given as privileged state(Ke et al., 2021), and typically demands an unsupervised intermediate representation and massive trajectory counts when applied to video(Li et al., 2020).Consequently, interaction structure in existing latent world models is usually hand-engineered or learned jointly under limited supervision, rather than discovered at scale.

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.



(a) Original scene **(b) Factored objects** **(c) Causal parents vs clutter**

Figure 1: A motivating example of object-centric causal analysis compared with (a) the original scene and (b) factored objects. (c) **OCCAM** can effectively identify the causal objects and clutters.

The lack of explicit causal structure becomes particularly problematic in complex, visually cluttered environments. Consider a multi-object manipulation task such as "push the plate to the front of the stove" in a kitchen scene containing task-irrelevant objects as shown in Figure 1. In a dense latent world model, the uninterpretable transition function intertwines the entire latent state, making predictions susceptible to spurious correlations from the background (a). Even if an object-centric representation grounds latent states into named entities (b), a fully connected transition network over the factored objects still permits unexplainable interactions between unrelated and task-irrelevant objects. Enforcing explicit causal sparsity (c) is essential to ensuring robust, interpretable predictions.

Motivated by this, we introduce the Object-Centric CAusal world Model (**OCCAM**), a framework that constructs language-grounded, explainable transition models. **OCCAM** leverages the semantic priors of vision-language models (VLMs), drawing upon their rich contextual knowledge and precise visual grounding capabilities. Specifically, we utilize open-vocabulary segmentation combined with SAM3 (Carion et al., 2026) tracking to name and extract all objects across episodes without manual mask annotations. Subsequently, rather than relying on data-hungry statistical causal discovery, **OCCAM** queries the VLM over motion-salient frames to infer a structural causal graph that identifies the causal relationship of each object. Eventually, the objects' bounding boxes are fed to a linear readout that shapes the encoder, while a causal transition network restricts the predictor such that an object's future prediction solely depends on its causal parents as derived from the causal graphs. To ensure statistical reliability, the VLM-proposed graph is rigorously validated against an independent, trajectory-based conditional mutual information (CMI) estimator inspired by Wang et al. (2022). **OCCAM** achieves explicit mechanistic transparency at a fixed, minimal annotation cost.

To systematically investigate the effectiveness of this framework, we hold the underlying architecture fixed and vary only the structural regime imposed on the transition function across a strict ablation ladder: dense, object-centric, and causal. Within each mode, structure is introduced progressively over a training curriculum rather than from the first step. We evaluate **OCCAM** along a complexity ladder spanning 2D manipulation (Push-T), 3D single-object control (OGBench-Cube), and 3D multi-object tasks (LIBERO-Goal), assessing **OCCAM** along four axes: goal-conditioned planning and control success, the object-centricity of the resulting representation, agreement between the VLM-proposed causal graph and an independently-estimated statistical reference, and how the benefit of explicit structure scales as scene complexity increases. The main contributions of this work can be summarized as follows:

- We introduce a latent world model equipped with explicit causal structure, evaluated through a strict architectural ablation ladder (dense, object-centric, and causal) that precisely isolates the impact of structural priors on transition dynamics.
- We propose a structural discovery pipeline that replaces both collapse-prone slot-attention-based object discovery and classical data-hungry causal discovery with a VLM-driven framework, grounding objects from natural language and extracting causal graphs at a low annotation cost, which are statistically validated post hoc with a CMI estimator.

- We empirically demonstrate across a domain complexity ladder that explicit causal structure preserves planning accuracy in simpler environments and achieves superior downstream control in complex, multi-object scenarios, providing an indicator that the value of explicit causal reasoning may scale with environment complexity.

2 Related Work

2.1 Dense world models

A large body of work learns world models densely: the entire state is mapped forward by a single, monolithic transition function, whether trained via reward-driven model-based RL (Hafner et al., 2019; 2020; 2024; Hansen et al., 2024) or via self-supervised prediction in a compact latent space (Zhou et al., 2025; Maes et al., 2026; Sobal et al., 2025). This design is simple and often highly performant, but gives the model no explicit notion of which entities exist in a scene or how they influence one another. Our dense mode corresponds to this family; LeWM (Maes et al., 2026) in particular is architecturally identical to it, since our dense mode inherits its encoder-predictor recipe and anti-collapse regularizer directly.

2.2 Object-centric and causal world models

A parallel line of work factors the latent into per-object representations, building on slot-based mechanisms for extracting exchangeable object representations from raw observations (Locatello et al., 2020; Kipf et al., 2022), motivated by the intuition that physical dynamics are naturally compositional over entities (Wu et al., 2023; Villar-Corrales et al., 2023). Without an explicit mechanism forcing the model to attend to inter-object dependencies, however, such models tend to fall back on modeling each object's self-dynamics or latch onto incidental correlations rather than genuine interactions. This has motivated work that explicitly targets interaction learning (Kipf et al., 2020; Feng et al., 2025; Mosbach et al., 2025), causal structure (Nishimoto & Matsubara, 2026; Zholus et al., 2022), or downstream-specific object-centric exploration (Ferraro et al., 2023), typically by separating self-dynamics from inter-object interaction terms architecturally (Mosbach et al., 2025; Villar-Corrales et al., 2023), regularizing attention to be sparse (Lei et al., 2025), or imposing an explicit graph structure by hand (Kipf et al., 2020; Zholus et al., 2022). An older line of structured world models represents the state as a set of interacting entities modeled with a relational or graph-based module (Kipf et al., 2020) or, alternatively, via pixel-reconstruction objectives (Lin et al., 2020; Veerapaneni et al., 2020), in either case operating outside a compact, encoder-derived latent space and limiting how directly these models compare to modern latent world models. C-JEPA (Nam et al., 2026) is closest in spirit to our object-centric and causal modes: it extends masked JEPA prediction to object-level tokens, using object-level masking as an implicit latent intervention leaving validation against ground-truth or statistically-derived causal graphs to future work, a gap we directly address with our CMI validator (§4.2).

Even when causal structure is the goal, discovering it remains a bottleneck. The broader causal representation learning literature (Schölkopf et al., 2021) shows this structure can be recovered under strong assumptions, such as access to known intervention targets over temporal sequences (Lippe et al., 2022). Classical causal discovery, whether via conditional-independence testing over hand-factored state (Spirtes et al., 2001; Wang et al., 2022; Zhu et al., 2022) or via video-based recovery requiring an unsupervised intermediate representation and many trajectories (Li et al., 2020), degrades once the variables of interest must first be extracted from raw pixels rather than given as privileged state (Ke et al., 2021). Overall, this work establishes that interaction structure matters for world modeling, but leaves open both how that structure should be obtained at scale, without the data and assumption burden classical discovery imposes, one of the gaps **OCCAM** is built to close.

3 Preliminaries

3.1 Problem Setting.

We learn from a fixed offline dataset of trajectories, each an episode of RGB observations o_t and actions $a_t \in \mathbb{R}^{d_a}$ for $t = 1, \ldots, T$. Each environment additionally has a natural-language task

description ℓ, a string available at training time but not part of the trajectory. We partition the action into K factored tokens, $a_t = [a_t^{(1)}, \ldots, a_t^{(K)}]$, rather than treating it as a single monolithic vector, because we want the causal structure we discover to be able to say which specific component of the action drives which object.

3.2 Latent World Models

We build our world model on the JEPA, since JEPA lets us train an encoder E_ϕ and a predictor P_θ to forecast future latents directly, without ever needing to decode back to pixels, which frees model capacity that would otherwise be spent reconstructing visual details irrelevant to control. Formally, E_ϕ maps each observation to a shared latent state:

$$z_t = E_\phi(o_t) \in \mathbb{R}^d, \tag{1}$$

and P_θ forecasts $\hat{z}_{t+1}$ from a history window of length H by minimizing a purely latent prediction loss $\mathcal{L}_{\text{pred}}$ against the true future latent. In its default form, P_θ is a single monolithic network, meaning every coordinate of $\hat{z}_{t+1}$ may depend on the entire conditioning latent z_t and the entire action a_t. We treat this default as our dense baseline precisely because it has no explicit notion of which entities exist in the scene or how they relate to one another, which is the exact property we want to isolate and vary.

3.3 Object-centric and causal factorization.

We let z_t give rise to a set of N per-object latent states:

$$S_t = \{s_t^{(1)}, \ldots, s_t^{(N)}\}, \qquad s_t^{(n)} \in \mathbb{R}^{d_o}, \tag{2}$$

where N is the number of objects tracked in a given environment, covering the full set of scene objects rather than a task-relevant subset, because we want the resulting structure to be tested on the same footing regardless of which objects a given task happens to mention. S_t is a further, object-level factorization derived from z_t. It is never an independent representation, since we want the object-centricity to be a property of the shared encoder itself. We write $\mathcal{I}(n) \subseteq \{1, \ldots, K\}$ for the subset of factored action tokens that are causal parents of object n, and represent the transition function's dependency structure as a graph over object indices:

$$\mathcal{G} = \{\, n : \mathrm{pa}(n) \,\}_{n=1}^{N}, \qquad \mathrm{pa}(n) \subseteq \big(\{1, \ldots, N\} \setminus \{n\}\big) \cup \{1, \ldots, K\}. \tag{3}$$

Under this notation, the dense predictor described above is simply the special case $\mathrm{pa}(n) = \{1, \ldots, N\} \setminus \{n\} \cup \{1, \ldots, K\}$ for every n: the implicit, fully-connected graph, expressed over z_t directly, since dense mode never factors the latent into objects at all.

Causal Validation. We adopt CDL (Wang et al., 2022)'s CMI estimation as our validation tool because it gives a precise definition to test the causality of our graph. CDL treats the environment as a structural causal model over the object states and the action: at each step, object i's next state is generated by:

$$s_i^{t+1} = f_i\big(\mathrm{pa}_{\mathcal{G}}(i),\, u_i^t\big), \tag{4}$$

with parents $\mathrm{pa}_{\mathcal{G}}(i) \subseteq \{s^t, a^t\}$ and mutually independent exogenous noise terms u_i^t. An edge $j \to i$ exists in $\mathcal{G}$ if and only if s_j^t is a direct cause of s_i^{t+1}, meaning an intervention $\mathrm{do}(s_j^t = x)$ shifts $p(s_i^{t+1} \mid \text{rest})$. Because the process is temporally ordered, we only ever need to decide whether an edge is present, not its direction, since all candidate edges necessarily point from t to $t{+}1$.

Under a standard set of identifiability conditions we state in full in the appendix, this graph is recoverable from purely observational trajectories through a single conditional-independence test per candidate edge: for any candidate parent j and target i:

$$(j \to i) \in \mathcal{G} \iff s_i^{t+1} \not\perp\!\!\!\perp s_j^t \mid \{s^t, a^t\} \setminus s_j^t, \tag{5}$$

equivalently the leave-one-out conditional mutual information $\mathrm{CMI}(j \to i) = I\big(s_i^{t+1}; s_j^t \mid \{s^t, a^t\} \setminus s_j^t\big) > 0$. We rely on this result because it means recovering the true parent set $\mathrm{pa}_{\mathcal{G}}(i)$ never requires a combinatorial search over graph structures. In practice, we estimate this CMI as a held-out predictive-error gap between a model conditioned on the full parent set and one with s_j removed, which we detail in the next section.

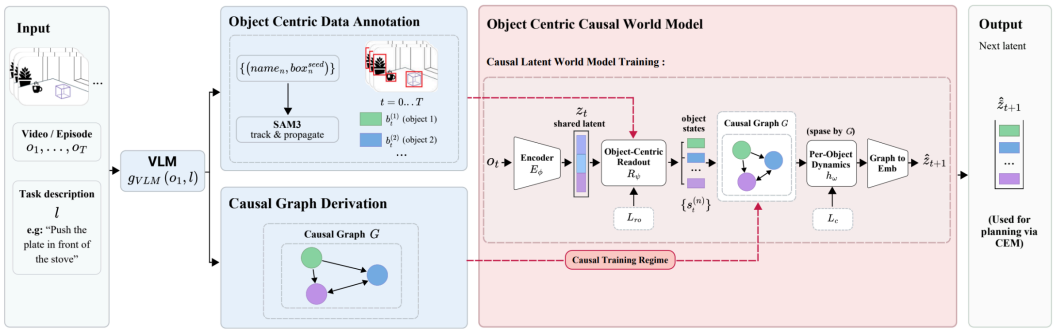


Figure 2: The **OCCAM** framework. A single vision-language model both grounds objects (OCDA) and proposes the causal graph $\mathcal{G}$; both feed a shared encoder driving a three-mode transition ladder. A conditional-mutual-information validator checks $\mathcal{G}$ offline against trajectory statistics.

4 METHOD

In this section, we detail our proposed framework, which comprises two primary components and a rigorous validation phase, as illustrated in Figure 2. The first component, Object-Centric Data Annotation (OCDA), extracts the required per-object supervision without the need for manual mask annotations. The second component entails the training of the causal latent world model, where this object-level supervision is integrated with a VLM-proposed causal graph. This training is conducted across three strictly comparable structural regimes to facilitate direct ablation. Finally, the validation phase corroborates the discovered causal graph against an independent statistical estimator.

4.1 OBJECT-CENTRIC DATA ANNOTATION

We build the OCDA to obtain object-level supervision without relying on hand-labeled masks, while keeping the annotation budget light. We prompt a vision-language model g_{VLM} with each episode's first frame o_1 alongside the task description ℓ, and it returns the list of objects in the scene:

$$g_{\text{VLM}}(o_1, \ell) \;\rightarrow\; \left\{(\text{name}_n,\ \text{box}_n^{\text{seed}})\right\}_{n=1}^{N}, \tag{6}$$

making sure to match the names the task description already provides where one exists, and assigning a consistent canonical name to any other object present but unmentioned by ℓ, along with an optional seed bounding box $\text{box}_n^{\text{seed}}$. Each name and optional seed box is then fed to SAM3, which grounds the corresponding object on the first frame and propagates the resulting mask through the trajectory frame by frame, producing at every step t a mask and a confidence score $c_t^{(n)} \in [0, 1]$; we take the normalized bounding box of this mask as our target $b_t^{(n)}$. We use $c_t^{(n)}$ to decide how much to trust it: we interpolate $b_t^{(n)}$ linearly from neighboring high-confidence frames wherever $c_t^{(n)}$ dips only briefly below threshold, and drop the episode for that object entirely once fewer than half of all frames meet the threshold, since majority-low-confidence tracking is more likely to reflect occlusion or outright failure than transient noise. This step is run over a small number of episodes per environment. At the end, OCDA produces the per-object bounding boxes $\{b_t^{(n)}\}$ that ground the causal latent world model's object-centric structure directly in the scene.

4.2 OBJECT-CENTRIC CAUSAL WORLD MODEL

We train our causal latent world model which consists of an encoder E_ϕ and an autoregressive Transformer predictor P_θ with AdaLN action conditioning, and we keep this encoder-predictor architecture fixed during the ablation with the other two regimes (dense, object-centric) to isolate the effect of each of the regimes.

We induce the object-centric factorization $S_t = \{s_t^{(n)}\}_{n \in \{1,\ldots,N\}}$ with a readout head R_ψ mapping the shared latent to each object's box, specifically the latent is factorized by a per-object linear



Figure 3: OCDA tracking on real episodes. Left: Push-T, pusher and T-block tracked from frame 0 to frame 54. Right: OGBench-Cube, robot and cube tracked from frame 0 to frame 100.

projection $W_n : s_t^{(n)} = W_n z_t$, into N object-specific subspaces, which we then decode by mapping to a normalized box prediction using a small head U_n and a sigmoid σ: $\hat{b}_t^{(n)} = \sigma(U_n s_t^{(n)})$. Writing $R_\psi^{(n)} = \sigma \circ U_n \circ W_n$, the readout is trained end to end using:

$$\mathcal{L}_{\text{ro}} = \sum_{n=1}^{N} \left\| R_\psi^{(n)}(z_t) - b_t^{(n)} \right\|_2^2, \tag{7}$$

with $b_t^{(n)}$ supplied by OCDA. With W_n linear and the gradient of R_ψ flowing back into E_ϕ, we shape the encoder to make z_t become object-decodable.

We query the VLM over a number of motion-salient event frames per environment, selected around moments of maximal inter-object displacement, and prompt it to propose the causal graph $\mathcal{G}$, after which we keep only the edges that appear on a majority of frames within an episode. We also make sure that the action is factorized into motion and gripper's action where applicable. Since the tracked object set is fixed across tasks, the VLM isn't restricted to task-relevant objects, so an edge irrelevant to one task gets discovered anyway and is available for other tasks.

Given the causal graph $\mathcal{G}$, we replace the dense predictor with a per-object network $h_\omega^{(n)}$ for the causal mode,

$$\hat{s}_{t+1}^{(n)} = h_\omega^{(n)}\Big(\big[\{s_t^{(m)}\}_{m \in \text{pa}(n) \cap \{1,\dots,N\}} \,\|\, \{a_t^{(k)}\}_{k \in \text{pa}(n) \cap \{1,\dots,K\}} \big] \Big), \tag{8}$$

and we restrict this network at construction to read only $\text{pa}(n)$, to ensure architecture-level sparsity. We then use a `graph_to_emb` head that re-lifts $\{\hat{s}_{t+1}^{(n)}\}_{n=1}^N$ back to the shared d-dimensional latent, to stay compatible with the same CEM rollout used by the other two modes, ensuring a fair comparison between all three during planning.

We introduce object-centric and causal structure only after the base representation has stabilized, at epochs e_{object} and e_{causal} respectively, since asking the model to satisfy an object or causal loss before it has learned a reasonable latent at all tends to destabilize training rather than help it:

$$\mathcal{L} = \mathcal{L}_{\text{pred}} + \lambda_{\text{sig}}\mathcal{L}_{\text{sig}} + \mathbb{1}[e \geq e_{\text{object}},\ \text{mode} \neq \text{dense}]\, \lambda_{\text{ro}}\mathcal{L}_{\text{ro}} + \mathbb{1}[e \geq e_{\text{causal}},\ \text{mode} = \text{causal}]\, \lambda_{\text{c}}\mathcal{L}_{\text{c}}. \tag{9}$$

Planning and evaluation protocols. Goal-conditioned CEM planning (Cube, Push-T) exercises P_θ directly: encode the goal image to $z_g = E_\phi(o_g)$, roll out candidate action sequences, minimize $\|\hat{z}_T - z_g\|_2^2$, identically across all three modes. LIBERO OFT-BC (Li et al., 2026; Kim et al., 2025) instead freezes E_ϕ and trains a lightweight behavior-cloning head on top, runs closed-loop, isolating encoder quality rather than predictor quality.

CMI validator. We validate $\mathcal{G}$ against a CMI estimator (§3.3), computed independently from trajectories; full derivation and thresholds are in Appendix F.1.

5 Experiments

We conduct various experiments to address the following questions: (1) How does **OCCAM** compare with other baseline models on planning and control success? (2) Does **OCCAM** lead to a more

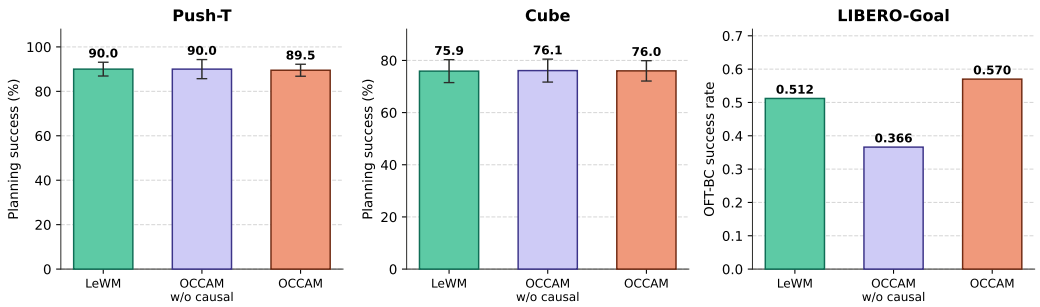


Figure 4: Planning and control success across all three environments (Push-T, Cube, LIBERO-Goal) for LeWM, OCCAM without causal structure, and OCCAM. Thick bars span mean ± std over 8 seeds (Push-T, Cube); the black tick marks the mean. LIBERO-Goal reports point-estimate OFT-BC success rate.

refined object-centric representation? (3) Do the VLM-proposed causal graphs agree with those obtained through statistical methods? (4) Does the benefit of causal structure scale in more complex environments?

5.1 Experimental setup

Environments. We evaluate OCCAM on three manipulation environments spanning a step in visual and physical complexity: Push-T (Chi et al., 2024), a 2D environment in which an agent pushes a T-shaped block into a target pose, contact-rich enough that small errors compound over a rollout; OGBench-Cube (Park et al., 2025), a 3D single-object environment in which a robot arm brings a cube to a target position via gripper-mediated interaction; and LIBERO-Goal (Liu et al., 2023), a 3D multi-object kitchen manipulation suite. This evaluation comprises 12 manipulation tasks in total: one each for Push-T and Cube, and the ten LIBERO-Goal tasks. We evaluate Push-T and Cube with goal-conditioned CEM planning, exercising the predictor directly, and LIBERO-Goal with a frozen-encoder OFT-BC protocol, isolating encoder quality instead.

Baselines. We compare against the three latent world models closest to our own architecture and training recipe. DINO-WM (Zhou et al., 2025) freezes a pretrained DINOv2 encoder and learns only a patch-token predictor, avoiding representation collapse at the cost of a large, encoder-fixed token budget. LeWM (Maes et al., 2026) is architecturally closest to our method. C-JEPA (Nam et al., 2026) extends masked JEPA prediction to object-level tokens, our closest object-centric point of comparison, though it reports Push-T only.

5.2 Results

Push-T and OGBench-Cube. On Push-T, **OCCAM** is statistically indistinguishable from LeWM (89.5 ± 2.7, 90.0 ± 3.1), and on Cube likewise (76.0 ± 3.9, 75.9 ± 4.4). **OCCAM**'s sparse, language-derived dependency structure does not trade away planning accuracy relative to the fully-connected baseline it replaces. Against the wider field, as shown in table 1, **OCCAM** is competitive with DINO-WM and roughly matches C-JEPA on Push-T. This establishes competitiveness with the state of the art; the finding this paper is built around is that structure is *free*, costing nothing while making the representation more interpretable.

Method	Push-T
DINO-WM	90
C-JEPA	88.67
OCCAM	89.5 ± 2.7

Table 1: Planning success rate (%) against baselines on Push-T. Baseline numbers are as reported in each paper's own evaluation; OCCAM reports mean ± std over 8 seeds.

LIBERO-Goal. As a stress test on a more complex, multi-object manipulation suite, we evaluate with an OFT-BC protocol: the encoder is frozen after world-model pretraining, a lightweight behavior-cloning head is trained on top for each task, and the resulting policy is run closed-loop

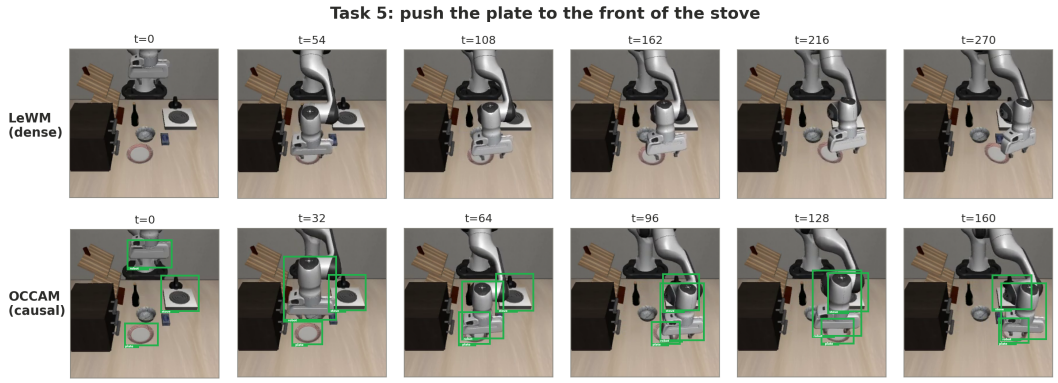


Figure 5: Rollout frames from LeWM (dense) and OCCAM (causal), task 5 (push the plate to the front of the stove), same initial state.

from deterministic initial states ($n_{\text{eval}} = 50$). Because both the encoder and the evaluation are deterministic, re-running the evaluation itself would not produce variance. Unlike in Li et al. (2026) where both the encoder and the action-chunk head are fine-tuned, we train only the lightweight BC head on the frozen encoder, which holds the representation fixed and therefore evaluates the encoder's representation quality. **OCCAM** outperforms LeWM (0.570 vs. 0.512), suggesting that the benefit of explicit structure may grow as scene complexity increases.

5.3 Analysis

Table 1 presents OCCAM's planning performance against DINO-WM and C-JEPA on Push-T. Figure 4 shows our own ablation modes across all three environments. OCCAM is competitive with the state of the art on Push-T, and within our own architecture, structure comes at no measurable cost regardless of environment. **Ablation ladder.** Within our own architecture, we isolate the effect of explicit structure by holding everything else fixed, and varying only the dependency structure imposed on the transition function across three modes: base LeWM, **OCCAM** w/o Causal, and **OCCAM** our causal mode. **OCCAM** w/o Causal and **OCCAM** only add supervision and structure on top of LeWM's base.

Qualitative rollout comparison. Figure 5 compares OCCAM's causal mode against LeWM's dense mode on task 5, which requires pushing the plate to the front of the stove. Under OCCAM, the arm goes directly to the plate, and partway through the push it visibly loosens its grip before re-establishing contact, allowing it to continue pushing the plate without flipping it. LeWM, in contrast, maintains a fixed grip throughout, which flips the plate and causes the arm to drift sideways into task-irrelevant space rather than completing the push.

On LIBERO-Goal, the overall LeWM $>$ **OCCAM** w/o Causal gap does not reflect object-centric structure being harmful in general: OCDA mis-grounded the bowl onto a visually similar distractor, and five of the ten tasks involve a bowl. Splitting by bowl-involving versus bowl-free tasks isolates the effect precisely for which we give more details in Appendix E.3:

Representation probing: object-centricity. A linear probe on the frozen $192-$d latent regresses each object's box center (Pearson r). Because per-model $r-$gaps are individually small, we establish significance with a within-object paired test that controls for object difficulty: each of the 9 probed objects (2 from Cube, 7 from LIBERO) contributes one LeWM-r and one **OCCAM**-r on identical frames, targets, probe and split.**OCCAM** is $\geq$ LeWM for every one of the 9 probed objects and strictly greater for $8/9$ (Only the LIBERO rack ties, already at 0.99 ceiling); a one-sided sign test gives $p = 0.004$ (Wilcoxon agrees). **OCCAM** w/o Causal exceeds LeWM on $7/9$ objects ($p = 0.008$). Push-T's near ceiling values all across all three modes reflect saturation: two salient movers are already at $r \approx 0.94$ under LeWM's dense baseline, leaving little room for structure to add measurable signal.

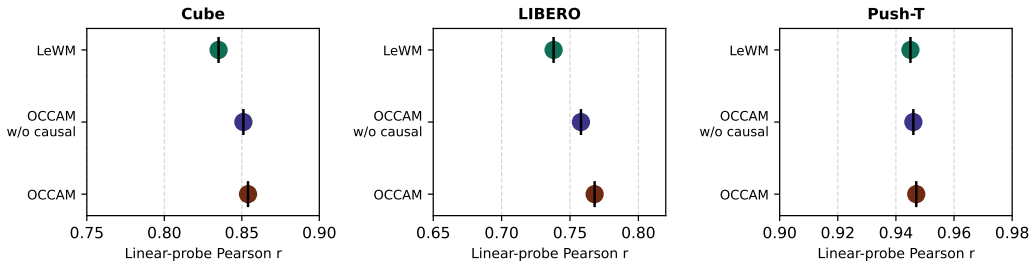


Figure 6: Linear-probe object-centricity (Pearson r) across Cube, LIBERO, and Push-T, for LeWM, OCCAM without causal structure, and OCCAM. The ordering LeWM $<$ OCCAM w/o Causal $<$ OCCAM holds on both multi-object environments; Push-T saturates near ceiling for all three modes.

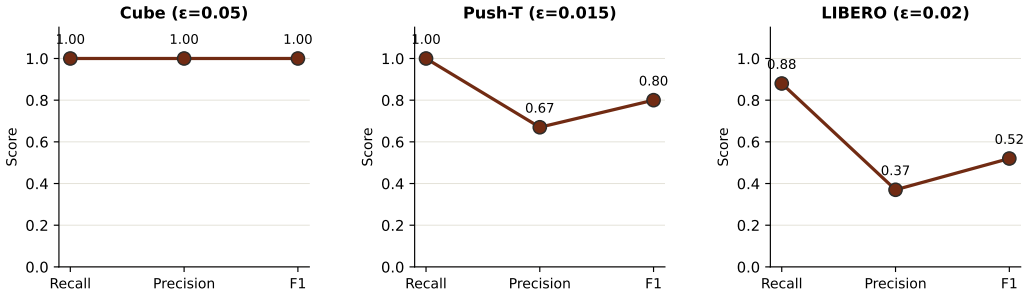


Figure 7: CMI validation of the VLM-proposed graph, per environment. Each panel traces recall, precision, and F1 against the independently-estimated CMI graph. Cube matches exactly on all three metrics; Push-T and LIBERO both show the same precision dip, consistent with the action-manipulator collinearity discussed in the text, with LIBERO's dip more pronounced.

CMI validation of the VLM-proposed graph. We estimate a conditional-mutual-information graph from OCDA object states and real actions and compare its edges to the VLM-proposed graph $\mathcal{G}$. As shown in Figure 7, for the Cube environment, the CMI graph matches $\mathcal{G}$ exactly. On the Push-T environment, both VLM edges are recovered (recall 1.0). The one false positive, action→T-block, is a mediated effect rather than a direct one. On the LIBERO environment, the task-independent union recovers all six robot→object edges plus bottle→rack (recall 0.88). The only missed edge is action→robot which is caused by a collinearity we explain further in Appendix E.4.

6 Conclusion and Limitation

OCCAM integrates language-grounded causal structure into JEPA transition functions without sacrificing planning accuracy, using a vision-language model for object grounding and causal graph generation alongside an independent CMI validator in place of data-hungry classical discovery. Across Push-T, OGBench-Cube, and LIBERO-Goal, **OCCAM** yields substantially more object-centric representations while matching baseline planning accuracy in simpler settings and achieving the strongest downstream control on the most complex, multi-object benchmark.

Limitation. **OCCAM**'s reliance on VLM grounding makes it susceptible to tracking failures, while causal relationship calculated by CMI validation inherits the identifiability limits of collinear variables; its centroid-based object state further omits orientation information. Future work includes online refinement of the causal graph during rollout, richer 3D object state representations, and evaluation under occlusion and in real-world settings.

AI USE STATEMENT

In this work, we used generative AI tools to assist with the writing of the manuscript text (eg: Grammar check, etc). We have not used generative AI tools to generate synthetic data, develop the paper's theoretical models or core hypotheses, assist in writing mathematical proofs, design or provide feedback on experimental methodology, implement the experimental methods, or interpret experimental results; all reported results were produced by our own training and evaluation runs. Assistance with translation, dataset cleaning/reformatting, and qualitative/thematic data analysis are not applicable to this work.

We have reviewed all AI-assisted work: all factual claims about prior work were cross-checked against the cited papers' own text, and all AI-drafted text were verified by the authors against our own logs. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Our work does not involve human subjects, personally identifiable data, or sensitive applications, and we don't identify any ethical concerns associated with this research. All benchmarks and models used in our experiments are publicly available and used in accordance with licenses.

REPRODUCIBILITY STATEMENT

Our implementation and experimental details are showcased in Appendix F. We will provide our code after acceptation.

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, 2023. URL https://arxiv.org/abs/2301.08243.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report, 2025. URL https://arxiv.org/abs/2511.21631.

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

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, Rishi Hazra, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollár, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. Sam 3: Segment anything with concepts, 2026. URL https://arxiv.org/abs/2511.16719.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion, 2024. URL https://arxiv.org/abs/2303.04137.

Fan Feng, Phillip Lippe, and Sara Magliacane. Learning interactive world model for object-centric reinforcement learning, 2025. URL https://arxiv.org/abs/2511.02225.

Stefano Ferraro, Pietro Mazzaglia, Tim Verbelen, and Bart Dhoedt. Focus: Object-centric world models for robotics manipulation, 2023. URL https://arxiv.org/abs/2307.02427.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels, 2019. URL https://arxiv.org/abs/1811.04551.

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

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models, 2024. URL https://arxiv.org/abs/2301.04104.

Nicklas Hansen, Hao Su, and Xiaolong Wang. Td-mpc2: Scalable, robust world models for continuous control, 2024. URL https://arxiv.org/abs/2310.16828.

Nan Rosemary Ke, Aniket Didolkar, Sarthak Mittal, Anirudh Goyal, Guillaume Lajoie, Stefan Bauer, Danilo Rezende, Yoshua Bengio, Michael Mozer, and Christopher Pal. Systematic evaluation of causal discovery in visual model based reinforcement learning, 2021. URL https://arxiv.org/abs/2107.00848.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success, 2025. URL https://arxiv.org/abs/2502.19645.

Thomas Kipf, Elise van der Pol, and Max Welling. Contrastive learning of structured world models, 2020. URL https://arxiv.org/abs/1911.12247.

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, 2022. URL https://arxiv.org/abs/2111.12594.

Yann LeCun. A path towards autonomous machine intelligence. *OpenReview preprint*, 2022.

Anson Lei, Bernhard Schölkopf, and Ingmar Posner. Spartan: A sparse transformer world model attending to what matters, 2025. URL https://arxiv.org/abs/2411.06890.

Wenyuan Li, Guang Li, Keisuke Maeda, Takahiro Ogawa, and Miki Haseyama. Predictive but not plannable: Rc-aux for latent world models, 2026. URL https://arxiv.org/abs/2605.07278.

Yunzhu Li, Antonio Torralba, Animashree Anandkumar, Dieter Fox, and Animesh Garg. Causal discovery in physical systems from videos, 2020. URL https://arxiv.org/abs/2007.00631.

Zhixuan Lin, Yi-Fu Wu, Skand Peri, Bofeng Fu, Jindong Jiang, and Sungjin Ahn. Improving generative imagination in object-centric world models, 2020. URL https://arxiv.org/abs/2010.02054.

Phillip Lippe, Sara Magliacane, Sindy Löwe, Yuki M. Asano, Taco Cohen, and Efstratios Gavves. Citris: Causal identifiability from temporal intervened sequences, 2022. URL https://arxiv.org/abs/2202.03169.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning, 2023. URL https://arxiv.org/abs/2306.03310.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention, 2020. URL https://arxiv.org/abs/2006.15055.

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

Malte Mosbach, Jan Niklas Ewertz, Angel Villar-Corrales, and Sven Behnke. Sold: Slot object-centric latent dynamics models for relational manipulation learning from pixels, 2025. URL https://arxiv.org/abs/2410.08822.

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, 2026. URL https://arxiv.org/abs/2511.14262.

Seohong Park, Kevin Frans, Benjamin Eysenbach, and Sergey Levine. Ogbench: Benchmarking offline goal-conditioned rl, 2025. URL https://arxiv.org/abs/2410.20092.

Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, New York, NY, USA, 2nd edition, 2009. ISBN 9780521895606. doi: 10.1017/CBO9780511803161.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Towards causal representation learning, 2021. URL https://arxiv.org/abs/2102.11107.

Vlad Sobal, Wancong Zhang, Kyunghyun Cho, Randall Balestriero, Tim G. J. Rudner, and Yann LeCun. Learning from reward-free offline data: A case for planning with latent dynamics models, 2025. URL https://arxiv.org/abs/2502.14819.

Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, Prediction, and Search*. The MIT Press, 2nd edition, 2001.

Rishi Veerapaneni, John D. Co-Reyes, Michael Chang, Michael Janner, Chelsea Finn, Jiajun Wu, Joshua B. Tenenbaum, and Sergey Levine. Entity abstraction in visual model-based reinforcement learning, 2020. URL https://arxiv.org/abs/1910.12827.

Angel Villar-Corrales, Ismail Wahdan, and Sven Behnke. Object-centric video prediction via decoupling of object dynamics and interactions, 2023. URL https://arxiv.org/abs/2302.11850.

Zizhao Wang, Xuesu Xiao, Zifan Xu, Yuke Zhu, and Peter Stone. Causal dynamics learning for task-independent state abstraction, 2022. URL https://arxiv.org/abs/2206.13452.

Ziyi Wu, Nikita Dvornik, Klaus Greff, Thomas Kipf, and Animesh Garg. Slotformer: Unsupervised visual dynamics simulation with object-centric models, 2023. URL https://arxiv.org/abs/2210.05861.

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. URL https://openreview.net/forum?id=BCGfDBOIcec.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. Dino-wm: World models on pre-trained visual features enable zero-shot planning, 2025. URL https://arxiv.org/abs/2411.04983.

Zheng-Mao Zhu, Xiong-Hui Chen, Hong-Long Tian, Kun Zhang, and Yang Yu. Offline reinforcement learning with causal structured world models, 2022. URL https://arxiv.org/abs/2206.01474.

A BROADER IMPACT

OCCAM is a research contribution aimed at making the internal structure of latent world models more interpretable and controllable, with manipulation-focused robotics as its primary evaluation domain. The immediate, foreseeable impacts are the standard dual-use considerations of robotic manipulation research generally, the same underlying capability (a world model that can plan contact-rich manipulation from vision) could be applied to beneficial uses (assistive robotics, manufacturing automation, warehouse logistics) or, less directly, to applications with more mixed societal impact (automation-driven labor displacement in manipulation-heavy industries). We do not see a path by which the specific contribution of this paper, explicit causal structure grounded via a VLM, validated statistically, meaningfully changes this risk profile relative to the dense world models it is compared against; if anything, the interpretability gains from an explicit, inspectable causal graph make model behavior easier to audit than an opaque, fully-connected transition function, which is a modest positive for safety and oversight. We do not release new data involving human subjects, and all environments used are existing public benchmarks.

B WORLD MODELS, JEPA, AND SIGREG

World models and latent-space planning. A world model learns an internal simulator $p(s_{t+1} \mid s_t, a_t)$ that an agent can query offline, via imagined rollouts, rather than acting in the real environment at every decision step. Model-based control built on such a model reduces to an optimization problem: find the action sequence whose imagined rollout best satisfies some objective (a reward, or, in the goal-conditioned case used throughout this paper, a distance to a goal state). The quality of the resulting plan is bounded by the fidelity of the world model itself, which motivates the central question this paper addresses: what should the world model's state representation, and the structure of its transition function, actually look like.

Latent (JEPA) vs. pixel-space world models. Reconstruction-based world models learn $p(o_{t+1} \mid o_t, a_t)$ directly over raw observations, typically via a generative decoder trained to minimize pixel-level reconstruction error. This has two costs: the model must spend capacity modeling irrelevant visual detail (texture, lighting, background clutter) that has no bearing on the dynamics relevant to control, and planning requires either decoding full frames at every rollout step or maintaining a separate latent dynamics model bolted onto the reconstruction objective. JEPA-style models sidestep both issues by predicting directly in a learned latent space: an encoder E_ϕ maps observations to a compact representation, and a predictor forecasts *future latents* rather than future pixels, with no decoder in the loop at all during planning. The tradeoff is that JEPA's training signal is weaker in one specific sense, there is no pixel-level ground truth to anchor the latent space, which opens the door to representational collapse (the trivial solution where the encoder maps every observation to the same constant vector, making the prediction loss trivially zero).

SIGReg: preventing collapse without negatives or momentum encoders. Prior collapse-avoidance strategies fall into a few families: contrastive objectives (which require negative samples and are sensitive to how those negatives are chosen), momentum/EMA target encoders (which double the parameter count and introduce a second set of hyperparameters governing the EMA decay), and stop-gradient asymmetric architectures (which work well empirically but lack a clean theoretical account of why they avoid collapse). SIGReg avoids all three by directly regularizing the *marginal distribution* of the embeddings toward a known, non-degenerate target, an isotropic Gaussian, using a distributional-distance test rather than a pairwise contrastive or asymmetric-architecture trick.

Concretely: for a batch of B embeddings $z \in \mathbb{R}^d$, draw K random unit-norm projection directions $\{\alpha_k\}_{k=1}^K$ and project each embedding onto each direction, $p = z^\top \alpha_k \in \mathbb{R}$. Under the null hypothesis that z is drawn from an isotropic Gaussian, every such 1-D projection p should itself be marginally standard normal, $p \sim \mathcal{N}(0, 1)$, regardless of which direction α_k was chosen. This is what makes a random-projection test tractable: rather than testing the full d-dimensional distribution directly, one tests many independent 1-D marginals and obtains an increasingly tight guarantee as K grows.

The 1-D test is based on the empirical characteristic function: for a target distribution with characteristic function $\phi(t) = \mathbb{E}[e^{itp}]$, a standard normal has the closed form $\phi(t) = e^{-t^2/2}$. SIGReg evaluates the gap between the empirical characteristic function of the batch's projected values and

this closed-form target, at a set of quadrature knots $\{t_j\}$ with weights $\{w_j\}$:

$$\mathcal{L}_{\text{sig}} = \frac{1}{K}\sum_{k=1}^{K}\sum_{j} w_j \Big[\big(\cos(t_j p) - \phi(t_j)\big)^2 + \big(\sin(t_j p)\big)^2\Big], \tag{10}$$

where $(\cdot)$ denotes an average over the batch. This loss is exactly zero if and only if every one of the K random projections is distributed as a standard normal, i.e., the batch's embeddings are isotropic Gaussian, giving a single differentiable regularizer with no negative sampling, no momentum encoder, and no stop-gradient asymmetry.

C Causality: Assumptions and Why CDL's CMI Test Is Valid

Structural causal model of the environment. We treat the environment as a structural causal model (SCM) over the object variables and the action. At each step, object i's next state is generated by $s_i^{t+1} = f_i(\mathrm{pa}_{\mathcal{G}}(i), u_i^t)$, with parents $\mathrm{pa}_{\mathcal{G}}(i) \subseteq \{s^t, a^t\}$ and mutually independent exogenous noise terms u_i^t. The dynamics graph $\mathcal{G}$ has an edge $j \to i$ iff s_j^t is a direct cause of s_i^{t+1}, an intervention $\mathrm{do}(s_j^t = x)$ shifts $p(s_i^{t+1} \mid \text{rest})$ (Pearl, 2009). Because the process is temporally ordered, all candidate edges point $t \to t{+}1$: unlike static causal discovery, which recovers only a Markov-equivalence class, the temporal ordering means we only need to decide edge presence, not direction.

Assumptions. We adopt the standard conditions under which $\mathcal{G}$ is identifiable from observational trajectories (Wang et al., 2022): (A1) the causal Markov condition and faithfulness (the trajectory distribution's conditional independencies are exactly those entailed by $\mathcal{G}$); (A2) Markovian, fully-observed dynamics ($\{s^t, a^t\}$ is a sufficient state); (A3) a self-loop $s_i^t \to s_i^{t+1}$ for every object; (A4) temporal acyclicity (no within-step or backward-in-time edges); (A5) factored transitions, $p(s^{t+1} \mid s^t, a^t) = \prod_i p(s_i^{t+1} \mid \mathrm{pa}_{\mathcal{G}}(i))$.

Proposition 1 (CMI identifiability; after Wang et al., 2022, Thm. 3.1). Under (A1)–(A5), for any candidate parent j and target i:

$$(j \to i) \in \mathcal{G} \iff s_i^{t+1} \not\perp\!\!\!\perp s_j^t \mid \{s^t, a^t\} \setminus s_j^t, \tag{11}$$

equivalently the leave-one-out conditional mutual information $\mathrm{CMI}(j \to i) = I\big(s_i^{t+1}; s_j^t \mid \{s^t, a^t\} \setminus s_j^t\big) > 0$. Thus a single conditional-independence test per candidate edge recovers $\mathrm{pa}_{\mathcal{G}}(i)$. We estimate this CMI as a held-out predictive-error gap between a full model and one with s_j removed (Appendix F.1).

What "causal" means here. Two assumptions are load-bearing and only partially checkable. Sufficiency (A2), that object centroids plus the action form a sufficient state, is an abstraction: a centroid omits, e.g., a block's orientation, so the recovered graph is causal with respect to the state we model, not necessarily the environment's full latent state. Faithfulness (A1) fails when a variable is a near-deterministic function of another (our action $\equiv$ manipulator velocity), making their individual contributions unidentifiable (§5.3). We therefore state the claim precisely: **OCCAM** recovers the trajectory-supported causal structure under (A1)–(A5), the causal graph identifiable from the observed object-and-action trajectories under the stated SCM abstraction, and we report exactly where (A1)/(A2) bind.

VLM and CMI as complementary evidence. The VLM proposes $\mathcal{G}$ from semantic and physical priors over a handful of event frames; CMI (Proposition 1) tests whether $\mathcal{G}$ is consistent with the trajectory distribution's conditional independencies. The two draw on different information, language priors vs. observed statistics, and could disagree, so agreement is a passed falsification test: evidence that $\mathcal{G}$ is the trajectory-supported causal structure, not a proof of the environment's true SCM. The same qualification applies to the VLM graph itself: it is a causal hypothesis, licensed as "causal" only under the SCM abstraction we validate.

D Algorithms

Here we present the OCDA's and the main framework **OCCAM**'s algorithms:

Algorithm 1 OCDA (Object-Centric Data Annotation)

Require: episode $\{o_1, \ldots, o_T\}$, task description ℓ, VLM g_{VLM}, segmenter SAM3

Ensure: per-object box targets $\{b_t^{(n)}\}$ for $n = 1..N$, $t = 1..T$

$\{(\text{name}_n, \text{box}_n^{\text{seed}})\}_{n=1}^{N} \leftarrow g_{\text{VLM}}(o_1, \ell)$ ▷ names + optionally seeds *every* visible object
for each object $n = 1, \ldots, N$ **do**
  $\text{mask}_1^{(n)}, c_1^{(n)} \leftarrow \text{SAM3.ground}(o_1, \text{name}_n, \text{box}_n^{\text{seed}})$
  for $t = 2, \ldots, T$ **do**
    $\text{mask}_t^{(n)}, c_t^{(n)} \leftarrow \text{SAM3.propagate}(o_t, \text{mask}_{t-1}^{(n)})$
  end for
  $b_t^{(n)} \leftarrow \text{normalized_bbox}(\text{mask}_t^{(n)})$ for all t
  if fraction of frames with $c_t^{(n)} <$ threshold < 0.5 **then**
    interpolate $b_t^{(n)}$ at low-confidence frames from neighbors
  else
    drop episode for object n entirely
  end if
end for
return $\{b_t^{(n)}\}$

Algorithm 2 Causal Latent World Model Training

Require: trajectories $\{(o_t, a_t)\}$, OCDA targets $\{b_t^{(n)}\}$, VLM graph $\mathcal{G}$, encoder E_ϕ, predictor P_θ, mode $\in \{\text{dense}, \text{object-centric}, \text{causal}\}$

$z_t \leftarrow E_\phi(o_t)$ ▷ shared latent
$\hat{z}_{t+1} \leftarrow P_\theta(z_{t-H+1:t}, a_{t-H+1:t})$ ▷ base JEPA prediction
$\mathcal{L} \leftarrow \|\hat{z}_{t+1} - z_{t+1}\|^2 + \lambda_{\text{sig}} \mathcal{L}_{\text{sig}}(z)$
if mode $\neq$ dense and epoch $\geq e_{\text{object}}$ **then**
  $S_t \leftarrow \text{readout}(z_t)$ ▷ per-object factorization
  $\mathcal{L} \leftarrow \mathcal{L} + \lambda_{\text{ro}} \sum_n \|R_\psi^{(n)}(z_t) - b_t^{(n)}\|^2$
end if
if mode $=$ causal and epoch $\geq e_{\text{causal}}$ **then**
  for each object n **do**
    $\hat{s}_{t+1}^{(n)} \leftarrow h_\omega^{(n)}\big([\{s_t^{(m)}\}_{m \in \text{pa}_\mathcal{G}(n)} \| \{a_t^{(k)}\}_{k \in \text{pa}_\mathcal{G}(n)}]\big)$
  end for
  $\mathcal{L} \leftarrow \mathcal{L} + \lambda_{\text{c}} \sum_n \|\hat{s}_{t+1}^{(n)} - s_{t+1}^{(n)}\|^2$
end if
backprop($\mathcal{L}$); update $\theta, \phi, \psi, \omega$

E MORE RESULTS

E.1 VLM-DERIVED GRAPH

We present the VLM derived graph for Libero-GOAL in Figure E.1.

E.2 LIBERO-GOAL PER-TASK RESULTS AND TASK DESCRIPTIONS

We read the contrast shown in task 5's results shown in figure 5 as consistent with two properties specific to the causal mode: object-centricity, since the arm remains engaged with the plate throughout rather than drifting toward irrelevant regions of the scene as the dense baseline does; and sparse, per-object attribution of the factored action tokens, since adjusting the grip has no effect on the plate's motion at that point in the push, a distinction the causal predictor's restricted parent sets can represent explicitly, while the dense predictor, which reads every action token for every object without restriction, cannot cleanly separate.

Figure 10 extends this comparison to task 8 (put the bowl on the plate) across two different initial states, one of the five bowl-involving tasks where OCDA's object grounding mis-latches onto a

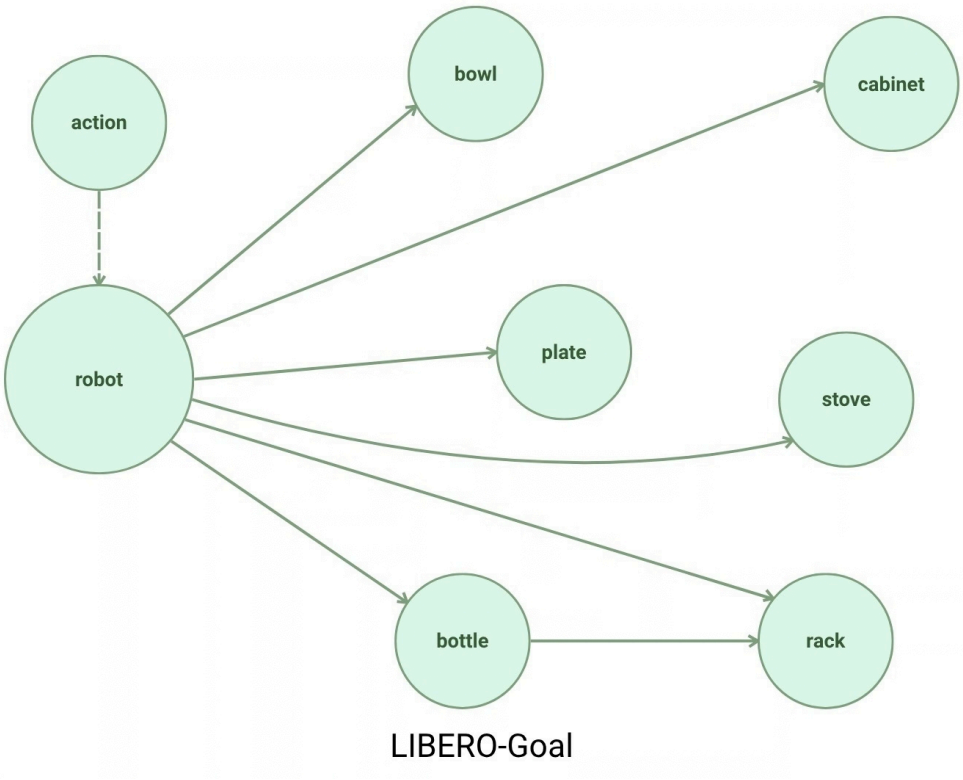


Figure 8: The VLM-proposed causal graph for LIBERO-Goal: a robot-centered star over six tracked objects (o1–o6, task-independent union across the ten-task suite) plus an isolated bottle→rack edge. CMI recovers seven of the eight VLM edges (recall 0.88): all six robot→object edges and bottle→rack. The only miss is action→robot (dashed), attributable to the same action-manipulator collinearity discussed below.

Task	Description	Bowl?
t0	Open the middle drawer of the cabinet	
t1	Put the bowl on the stove	✓
t2	Put the wine bottle on the cabinet	
t3	Open the top drawer and put the bowl inside	✓
t4	Put the bowl on top of the cabinet	✓
t5	Push the plate to the front of the stove	
t6	Put the cream cheese in the bowl	✓
t7	Turn on the stove	
t8	Put the bowl on the plate	✓
t9	Put the wine bottle on the rack	

Table 2: LIBERO-Goal task descriptions and bowl involvement. Five of the ten tasks (t1, t3, t4, t6, t8) involve the bowl object that OCDA mis-grounded onto a white distractor (§5.3).

visually similar distractor. In the first initial state, OCCAM's arm moves directly to the region immediately around the bowl and completes the task despite the mis-grounding, consistent with the causal mode's sparsity containing the effect of the bowl's corrupted representation to the bowl's own prediction rather than letting it propagate into the plate's and the robot's, which OCDA grounds correctly; this is a guarantee object-centric structure alone does not provide, since a fully-connected predictor lets one object's bad representation contaminate every other object's prediction as well. LeWM's dense mode instead maintains its grip past the point of completion, ultimately flipping

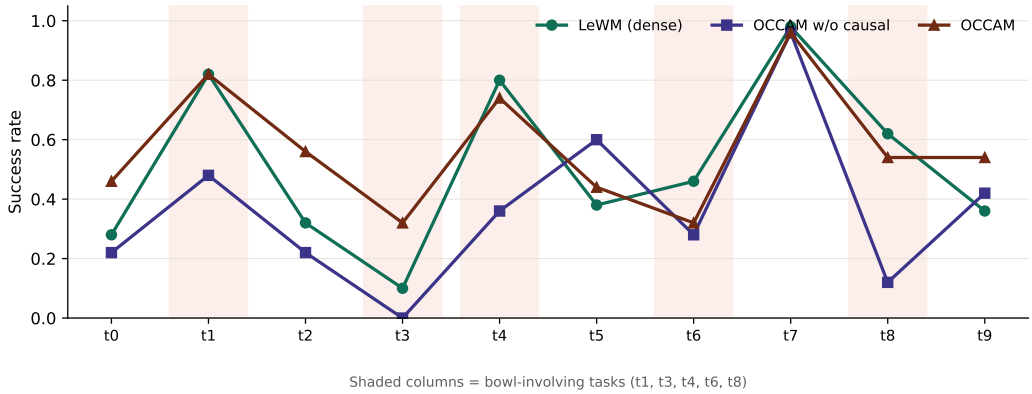


Figure 9: LIBERO-Goal per-task success rate ($n = 50$) for LeWM, OCCAM without causal structure, and OCCAM. Shaded columns mark the five bowl-involving tasks (t1, t3, t4, t6, t8), where OCDA's grounding failure is concentrated. OCCAM without causal structure visibly drops below both other modes specifically within the shaded region; OCCAM remains close to LeWM throughout and is the strongest mode on the unshaded, bowl-free tasks.

Subset	LeWM	**OCCAM** w/o Causal	**OCCAM**
Bowl tasks (5)	0.560	0.248	0.548
Non-bowl tasks (5)	0.464	0.484	0.592

Table 3: LIBERO-Goal success split by whether the task involves the mis-grounded bowl object. **OCCAM** w/o Causal's collapse is confined entirely to the bowl subset; on bowl-free tasks it already matches or exceeds LeWM, and **OCCAM** remains consistently strong.

both the plate and the bowl. In the second initial state, OCCAM again completes the task cleanly, while LeWM's arm never reaches the bowl at all, instead hovering and circling nearby, a failure mode consistent with the dense predictor's lack of object-centric structure allowing task-irrelevant features in the scene to dominate its behavior.

E.3 Bowl tasks ablation

On bowl tasks, **OCCAM** w/o Causal collapses to 0.248 while LeWM and **OCCAM** remain close (0.560 and 0.548 respectively, **OCCAM** is robust to the bad node). On the five bowl-free tasks, where grounding is unaffected, **OCCAM** w/o Causal is already at least on par with LeWM (0.484 vs 0.464), and **OCCAM** is clearly the best mode of the three (0.592). This localizes the overall LeWM $>$ **OCCAM** w/o Causal ordering entirely to a single mis-grounded node rather than to object-centric structure itself, and shows **OCCAM**'s advantage is not merely resilience to that failure: on the unaffected tasks it is the strongest mode outright.

E.4 Collinearity explanation.

The action is essentially the manipulator's velocity command, so action and manipulator-velocity are near-collinear in the data. CDL's leave-one-out test splits credit between action→object and manipulator→object when both carry nearly the same information, which simultaneously (i) suppresses the true action→manipulator edge and (ii) spawns the mediated action→object edges noted above, a faithfulness-boundary identifiability limit of the CMI estimator itself, not an error in the VLM graph. We confirm this directly: the pusher→T-block edge scores CMI $= 0.023$ conditioned on $\{$T-block, action$\}$, versus CMI $= 0.198$ conditioned on $\{$T-block$\}$ alone, a $9\times$ increase once the action is dropped from the conditioning set, confirming that conditioning on the physical state exposes the true edge.

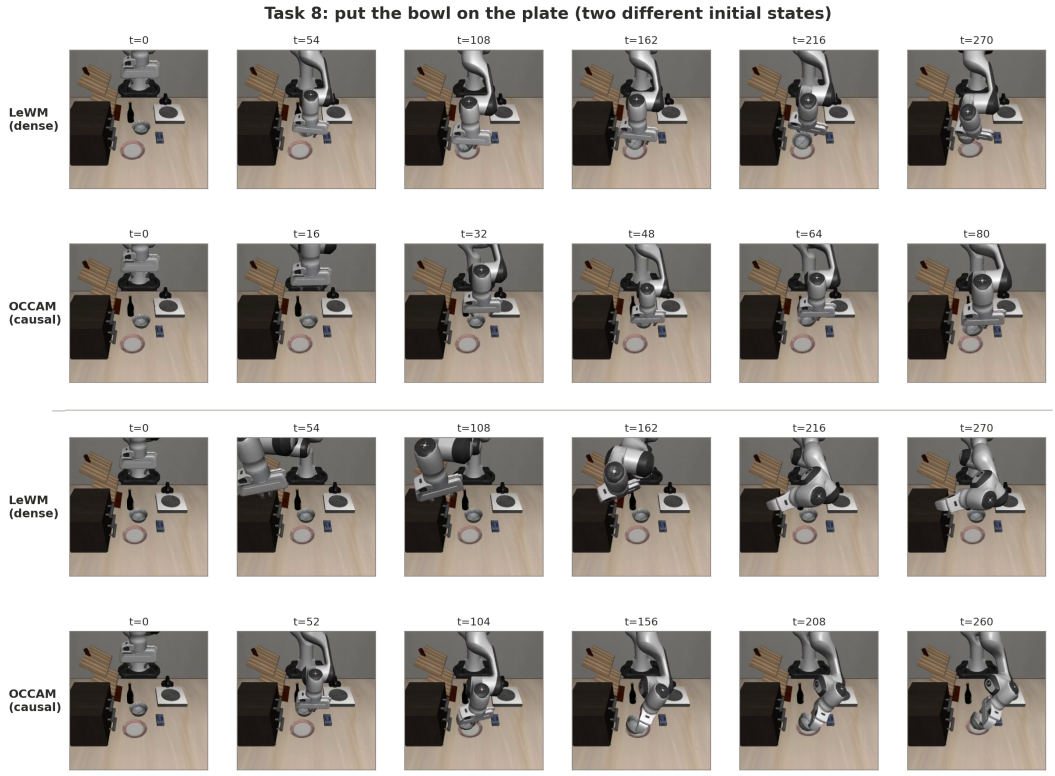


Figure 10: Rollout frames from LeWM (dense) and OCCAM (causal), task 8 (put the bowl on the plate), across two different initial states; one of the five bowl-involving tasks where OCDA's object grounding mis-latches onto a visually similar distractor. In the first initial state, LeWM maintains its grip past task completion and flips both the plate and the bowl, while OCCAM completes the task and stops. In the second, LeWM's arm never reaches the bowl and instead hovers and circles nearby, while OCCAM again completes the task cleanly.

E.5 Frameskip ablation

The world-model timestep is a first-order design choice on contact-rich dynamics: Push-T dense success rises from 43% (fs=1) to 90% (fs=5), as a fixed lookahead needs $5\times$ fewer model steps and thus less compounding drift, whereas near-Markovian Cube is frameskip-robust (74%→76%). We use fs=5 throughout the main results.

F Experimental Details

F.1 Full CMI implementation details

Estimator. We estimate the reference causal graph with the predictor-based conditional-mutual-information (CMI) test of Causal Dynamics Learning (CDL; Wang et al., 2022). For a target variable i and candidate parent j, we train two MLP predictors of the standardized H-step state delta Δs_i: a full model f_{full} conditioned on all object states plus the action, and a leave-one-out model f_{-j} with j removed. As in CDL, $\mathrm{CMI}(j \to i) = \mathrm{MSE}(f_{-j}) - \mathrm{MSE}(f_{\text{full}})$ on a held-out split, and an edge is accepted iff $\mathrm{CMI} > \varepsilon$. We differ from CDL only in using separate MLPs per conditioning set (rather than one input-masked network), predicting the standardized state delta (not the absolute next state), and estimating at the world-model timestep $H =$ frameskip.

Preprocessing. A physical filter drops transitions with position-delta or velocity magnitude > 0.3 (teleports/boundary jumps); an episode-quality filter drops episodes where an object shrinks below

Component	Setting
Architecture	MLP, 2×128, ReLU
Optimizer	Adam, lr 10^{-3}
Batch / epochs	256 / 200
Train/val	80/20 (CMI on val MSE)
Target	z-scored H-step Δcentroid
Inputs	centroid + velocity (+ action)

Table 4: CMI predictor network and training, identical across environments.

Env	ε	H	seeds	node state
Cube	0.05	5	1	centroid (2)
Push-T	0.015	5	1	centroid (2)
LIBERO	0.02	1	5	centroid + proprio (11)

Table 5: CMI per-environment settings.

$0.3\times$ its own median area (lost track) or overlaps the goal for > 0.3 of frames (mistracked). For LIBERO the estimate is averaged over 5 seeds per task and combined into one task-independent graph by union with $\geq$50% fraction voting.

No-action ablation (Push-T). $\mathrm{CMI}(\text{pusher} \to \text{T-block} \mid \text{T-block}, \text{action}) = 0.023$ vs. $\mathrm{CMI}(\text{pusher} \to \text{T-block} \mid \text{T-block}) = 0.198$, a $9\times$ increase once the action is dropped from the conditioning set, direct evidence of the action-manipulator collinearity discussed in Appendix C and §5.3.

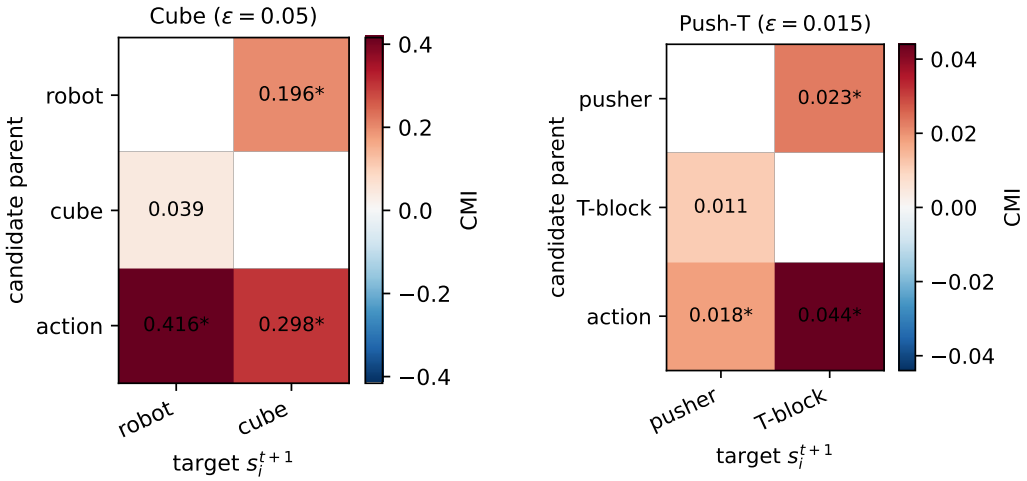


Figure 11: CMI score matrices (candidate parent $\times$ target), Cube (left) and Push-T (right). Starred cells exceed ε and are accepted edges.

F.2 Architecture, hyperparameters, and dataset details

Shared backbone. From-scratch ViT-tiny encoder (192-d CLS token, patch size 14), autoregressive Transformer predictor (depth 6, heads 16), history window $H = 3$, AdamW at learning rate 5×10^{-5}, 10 training epochs for PushT and OGCube (40 epochs for LIBERO-Goal), SIGReg weight $\lambda_{\text{sig}} = 0.09$.

e_{object}, e_{causal}, λ_{ro}, λ_{c}, values used were $e_{\text{object}} = 3$, $e_{\text{causal}} = 7$, $\lambda_{\text{ro}} = 0.1$, $\lambda_{\text{c}} = 0.05$

Datasets. Cube, OGBench `cube_expert`, 500 episodes, ∼100k frames, raw/effective action dim 5/25 at frameskip 5. Push-T, expert corpus, 18,685 episodes, ∼1.9M frames, raw/effective action dim 2/10 at frameskip 5. LIBERO-Goal, agentview camera, 63,728 frames across the 10-task suite. OCDA-annotated episode counts: 500 (Cube), 658 good episodes of 799 attempted (Push-T, ∼82% yield after confidence filtering), a subset for LIBERO. A Push-T ablation at reduced data (500 episodes) collapses dense-mode planning to 0%, isolating data quantity, not architecture, as the Push-T performance driver at low data.

F.3 VLM MODEL AND PROMPTS

Model. Both object grounding (OCDA) and causal-graph discovery use Qwen3-VL-8B, queried in two configurations: **Qwen3-VL-8B-Instruct** and **Qwen3-VL-8B-Thinking** (Bai et al., 2025).

Object grounding, per environment. The VLM is shown the first frame together with the task description and asked to identify every visible object via open per-frame detection, not to locate a fixed expected list, this is what lets OCDA track distractor/background objects beyond whatever the task description names. Example canonical names produced: Push-T, "pusher," "T-block," "goal." Cube, "robot," "cube." LIBERO-Goal, task-dependent (e.g. "bowl," "plate," "wine bottle," "rack," "stove," "cabinet drawer," "robot"), with the bowl being the one case where grounding mis-latched onto a visually similar white distractor (§5.3).

Causal graph discovery. After objects are grounded and named, the VLM is queried over motion-salient event frames with a prompt of the following form (abridged):

```
You are analyzing ONE manipulation episode shown as an
ORDERED sequence of n frames.  For each frame I also give
the commanded action magnitude(s).  Based ONLY on what you
SEE change across the frames, decide the DIRECT causal
influences ("direct" = no other listed entity mediates
it):  (a) action→object (is the object's change directly
driven by the commanded action?); (b) object→object (does
one object's motion directly drive another's?).  Output only
edges with a confidence, as JSON.
```

Edges are kept by majority vote within an episode and unioned across episodes; parents may be objects or factored action tokens (`action:motion`, `action:gripper`), so the action is never treated as a single monolithic cause.