

FactorJEPA: Factorizing Monolithic Futures into Layout-Agent-Interaction Channels for Crowded and Chaotic Global South Urban Worlds

Kapil Wanaskar^{1*}, Gaytri Jena², Aman Chadha³, Vinija Jain⁴, Vasu Sharma⁵, Amitava Das⁶

¹San Jose State University, USA ²UC Berkeley, USA ³Apple, USA ⁴Meta, USA ⁵PocketFM, USA ⁶Pragya Lab, BITS Pilani Goa, India

Abstract

World models have attracted significant attention for their ability to capture and predict the **structure and dynamics of the physical world**. In this emerging landscape, **Joint Embedding Predictive Architectures (JEPA)** offer a particularly compelling direction.

In this paper, we study a largely unexplored regime: **populous, crowded, and chaotic Global South urban environments**, which we call **DENSEWORLD**. Unlike the lower-density, lane-structured settings that dominate existing evaluations, these scenes exhibit **soft spatial boundaries**, **extreme agent heterogeneity**, **persistent occlusion**, and **rapid social negotiation** under mixed traffic. We introduce the first large-scale dataset for this regime, comprising **1,000 hours of drive-through, walk-through, and aerial video collected across 22 cities**. The dataset covers a wide variety of scene types, times of day, weather conditions, crowd and traffic densities, traffic mixes, pedestrian-vehicle separation patterns, road layouts and surfaces, infrastructure quality, encroachment levels, locally distinctive objects, vegetation, lighting conditions, and video quality. Our evaluation reveals that existing JEPA formulations struggle to preserve dense interaction dynamics under heterogeneity and partial observability.

We introduce **FactorJEPA**, which makes **world structure a first-class predictive primitive**. Rather than encoding the future in a monolithic latent, it composes **layout, entities, and interactions**, using a visibility gate and separated subspaces to preserve partially observed agents and discourage cross-factor shortcuts. FactorJEPA improves (i) **future-latent accuracy** (Future-frame L1), (ii) **intervention-sensitive prediction** (Causal L1), and (iii) **robustness to reduced visual evidence** (Mask-ratio slope), while exposing (iv) **a reproducible motion-information trade-off** (Motion cosine). Method rankings replicate across 2B and 1B V-JEPA 2.1 backbones, with $\rho = 0.895\text{--}0.978$ across these four diagnostics.

Data and models. We publicly release the [DENSEWORLD-115k dataset](#) and the surgery-trained [FactorJEPA checkpoints](#).

DENSEWORLD: A Benchmark for Populous, Crowded, and Chaotic Global South

We use **DENSEWORLD** to denote a **critical but under-represented world-modeling regime: populous, crowded,**

*All authors conducted this work independently, outside their roles and employment at their respective companies.

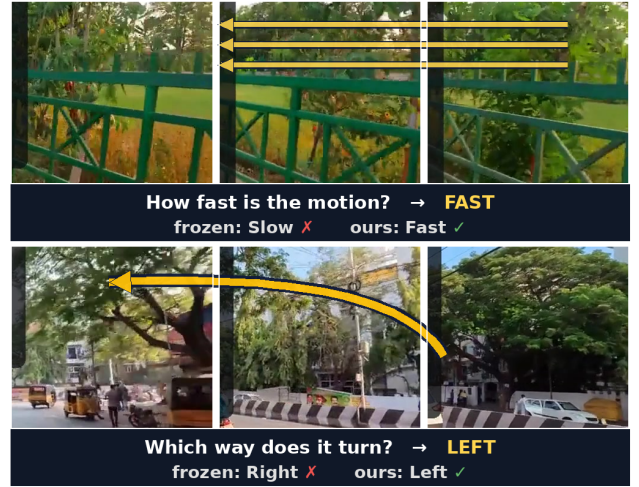


Figure 1: **Same question, two encoders.** Each row is a three-frame filmstrip of one held-out clip posed as a multiple-choice question, answered from an identical probe head over the frozen V-JEPA 2.1 encoder vs. ours (factor-view predictor surgery); only the backbone differs. *Top*: motion speed (ours 69.8% vs frozen 60.9%). *Bottom*: turn direction.

and chaotic Global South urban scenes. It is defined not only by geography, but by measurable properties: **high agent count density**, **high agent occupancy**, **persistent occlusion**, **agent heterogeneity**, and **interaction pressure**. This regime matters because many emerging urban environments combine **high population density**, **mixed mobility systems**, **diverse infrastructure patterns**, and **rapid spatial change**, making them scientifically important yet underrepresented in world-model benchmarks (Mahendra et al. 2021; Ellis and Roberts 2016). Thus, world models must handle uncertainty, density, and relational complexity that is less prominent in lower-density, lane-structured benchmarks. See Figure 2 for representative DENSEWORLD scenes.

We characterize DENSEWORLD through four coupled properties. **First**, it exhibits **soft spatial boundaries**: road edges, drivable space, and pedestrian regions are often negotiated rather than crisply marked (Varma et al. 2019; Dokania et al. 2023). **Second**, it contains **extreme agent hetero-**



Figure 2: **DENSEWORLD 1.0 scene-type coverage.** Representative examples from the dataset illustrate the breadth of urban environments covered by DENSEWORLD 1.0, including **market, residential, commercial, promenade, transit, highway, heritage, junction, flyover, and beach** scenes. This diversity reflects the spatial, social, and infrastructural heterogeneity of populous, crowded, and chaotic urban environments.

geneity, with pedestrians, cars, two-wheelers, carts, animals, delivery vehicles, and rickshaws coexist locally (Khan and Maini 1999; Asaithambi et al. 2012). **Third**, it has **persistent occlusion** from density, clutter, and near-field encounters, producing severe **partial observability** (Varma et al. 2019; Dokania et al. 2023); prediction must rely on *memory, object permanence*, and *relational inference*, not only visible appearance. **Fourth**, motion is governed less by rigid rule compliance than by rapid **social negotiation** under mixed traffic and weak lane discipline (Papathanasopoulou and Antoniou 2018; Asaithambi et al. 2012).

Together, these properties make DENSEWORLD a **natural stress test** for world models: success requires representations that preserve **layout, agent state, and interaction dynamics** under uncertainty, rather than collapsing into one **entangled predictive latent**. As shown in Section , we measure this regime through agent count density, agent occupancy, occlusion pressure, interaction pressure, and agent heterogeneity. This motivates a predictor that separates **layout, entities, and interactions**.

DENSEWORLD 1.0: the Benchmark

We partnered with two professional video-collection companies to record approximately **1,000 hours** of urban footage across **22 Tier-1 and Tier-2 cities in India**. The data spans **drive-through, walk-through, and aerial** viewpoints, and long-form recordings are segmented using scene-detection and shot-boundary methods into clips of **6–13 seconds**. DENSEWORLD 1.0 covers commercial, residential, transit, heritage, coastal, and high-density street settings, including **markets, commercial streets, transit corridors, junctions, flyovers, bazaars, ghats, beaches, and skylines**. It further spans variations in the **time of day, weather, crowd density, traffic mix, pedestrian–vehicle separation, road geometry, surface quality, encroachment, and lighting**. Beyond standard urban actors, the dataset includes locally frequent agents and objects such as **auto-rickshaws, cycle rickshaws, street vendors, and animals**. Before release, all footage is processed through privacy-preserving filters, including face

and license-plate blurring and removal of sensitive segments.

A defining feature of DENSEWORLD is **heterogeneous mixed traffic**: diverse agents share the same right of way under **weak lane discipline**, creating strong lateral and longitudinal coupling (Khan and Maini 1999; Papathanasopoulou and Antoniou 2018; Asaithambi et al. 2012). Unlike lane-structured settings, these scenes exhibit **fluid spatial support**, unstable visibility, and dense **local negotiation** among cars, buses, trucks, auto-rickshaws, two-wheelers, bicycles, carts, and pedestrians.

How Dense is DENSEWORLD?

To make this regime measurable, we compare DENSEWORLD against **BDD100K** and **nuScenes**, two widely used benchmarks for autonomous-driving perception and scene understanding (Yu et al. 2020; Caesar et al. 2020). The goal is not to claim that DENSEWORLD scenes merely look different, but to test whether they occupy a quantitatively distinct operating regime for world-model evaluation.

We measure this regime along five axes: **agent count density, agent occupancy, occlusion pressure, interaction pressure, and agent heterogeneity**. These metrics capture, respectively, the number of dynamic agents, the fraction of image area they occupy, the frequency of partial or heavy occlusion, the density of spatially proximate agent pairs, and the diversity of co-occurring actor categories. Together, they quantify the multi-agent load, visual congestion, visibility degradation, local interaction structure, and traffic diversity of each scene.

For an auditable comparison, all statistics are computed after mapping labels to a **shared dynamic-agent taxonomy** and matching comparable scene categories across datasets. Figure 3 shows that DENSEWORLD has consistently higher density and occupancy across matched categories, with the largest gaps in **market, commercial, and flyover/underpass** scenes. These results support the central premise of the benchmark: DENSEWORLD is not merely a geographic extension of existing driving data; it defines a higher-density, higher-occlusion, and higher-interaction regime for evaluat-

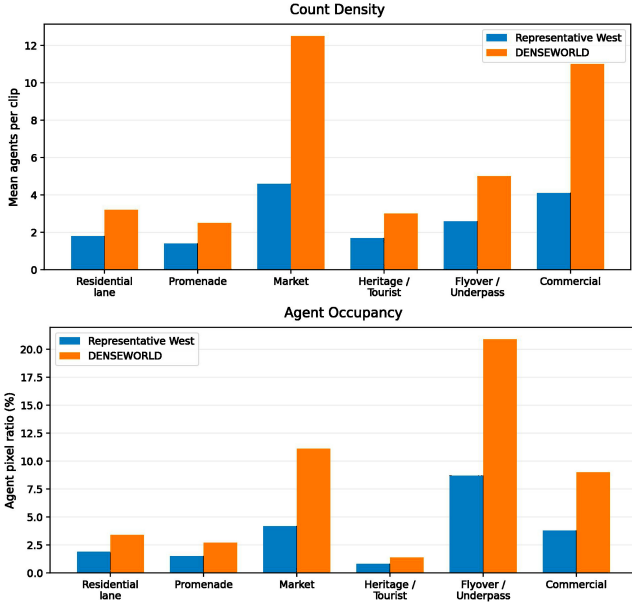


Figure 3: **DENSEWORLD** exhibits higher multi-agent density than standard driving benchmarks. We compare agent count density and agent occupancy across matched scene types. DENSEWORLD shows large gaps in **market**, **commercial**, and **flyover/underpass** scenes, indicating stronger interaction pressure and heavier visual competition.

ing predictive world models.

Can Fine-Tuning Close the DENSEWORLD Gap?

Before introducing FactorJEPA, we ask whether conventional adaptation can recover the predictive structure required by DENSEWORLD without reorganizing the JEPA predictor. We evaluate three complementary strategies: **LoRA** (Hu et al. 2022), **DoRA** (Liu et al. 2024), and **Auto-RGN** (Lee et al. 2023). All use the same raw clips, optimization steps, optimizer, masking policy, and evaluation protocol; only the parameter-update mechanism changes.

Off-the-shelf frozen encoders collapse into a narrow 37.5–44.4% band on the motion probe (Table 1), and no single model wins every column, so frozen reuse is not enough.

Let x denote a video clip; M_c and M_t the context and target masks; f_θ the online encoder; $\bar{f}_\theta$ the momentum target encoder; and g_ϕ the predictor. All baselines retain the executed V-JEPA objective:

$$\mathcal{L}_{\text{JEPA}} = \mathbb{E}_x \left[\left\| g_\phi(f_\theta(M_c \odot x), M_t) - \text{sg}(\bar{f}_\theta(M_t \odot x)) \right\|_2^2 \right],$$

where sg denotes stop-gradient. This data- and step-matched protocol isolates adaptation capacity without introducing a different prediction target or training signal.

For LoRA, an adapted weight matrix is

$$W' = W_0 + \frac{\alpha}{r} BA, \quad B \in \mathbb{R}^{d_{\text{out}} \times r}, \quad A \in \mathbb{R}^{r \times d_{\text{in}}},$$

where $r \ll \min(d_{\text{in}}, d_{\text{out}})$. DoRA further separates the magnitude and direction of each weight vector:

$$W' = m \odot \frac{W_0 + \Delta W}{\|W_0 + \Delta W\|}, \quad \Delta W = \frac{\alpha}{r} BA,$$

allowing directional adaptation without coupling it to weight magnitude.

Following the relative-gradient-norm criterion of Lee et al. (2023), our blockwise Auto-RGN implementation scores transformer block ℓ as

$$s_\ell = \frac{\|\nabla_{\theta_\ell} \mathcal{L}_{\text{JEPA}}\|_2}{\|\theta_\ell\|_2 + \epsilon}, \quad \mathcal{S}_K = \text{TopK}_\ell(s_\ell),$$

and updates only $\{\theta_\ell : \ell \in \mathcal{S}_K\}$. Normalization by parameter magnitude prevents larger blocks from being favored solely because of scale. LoRA and DoRA target the same projection families, while Auto-RGN receives a matched trainable-parameter budget.

FactorJEPA: Explicitly Factorized Predictive Channels

Motivation. Let x be a video clip, M_c and M_t its context and target masks, f_θ the online encoder, and $\bar{f}_\theta$ the momentum target encoder. The context representation and stop-gradient future target are

$$h_t = f_\theta(M_c \odot x), \quad Y_{t+\Delta}^* = \text{sg}(\Pi_{M_t}[\bar{f}_\theta(x)]) \in \mathbb{R}^{m \times d}$$

where Π_{M_t} denotes the executed target-selection operator, m the number of target representations, and d their embedding dimension. A conventional JEPA predictor learns

$$g_\phi : (h_t, M_t) \mapsto \hat{Y}_{t+\Delta} \in \mathbb{R}^{m \times d}$$

by matching $\hat{Y}_{t+\Delta}$ to $Y_{t+\Delta}^*$. The objective specifies *what* to predict, but leaves *how predictive information is internally organized* unconstrained.

This ambiguity is consequential in **DENSEWORLD**, where layout, agent density, visibility, and interaction pressure are strongly correlated. A high-capacity predictor can exploit **shortcut mixtures**—crowd texture as a proxy for interaction, visible appearance for object persistence, or road geometry for motion—without recovering the structure governing scene evolution.

FactorJEPA resolves this ambiguity by replacing the monolithic predictor with explicit **layout**, **agent**, and **interaction** channels. A **soft visibility gate** attenuates uncertain observations, while block-structured regularization limits **cross-factor leakage**. Together, these components factorize the future JEPA embedding into semantically anchored coordinates and factor-specific predictive subspaces.

DINOv2-Based Segmentations

Agent-Layout-Interactions

For each privacy-filtered clip x_n , a frozen DINOv2 pipeline (Oquab et al. 2024) extracts region masks, boxes, descriptors, and confidences, which are temporally associated into tracklets:

$$\mathcal{P}_n = \mathcal{D}_{\text{pre}}(\mathcal{F}_{\text{DINOv2}}(x_n)), \quad \mathcal{T}_n = \mathcal{A}(\mathcal{P}_n).$$

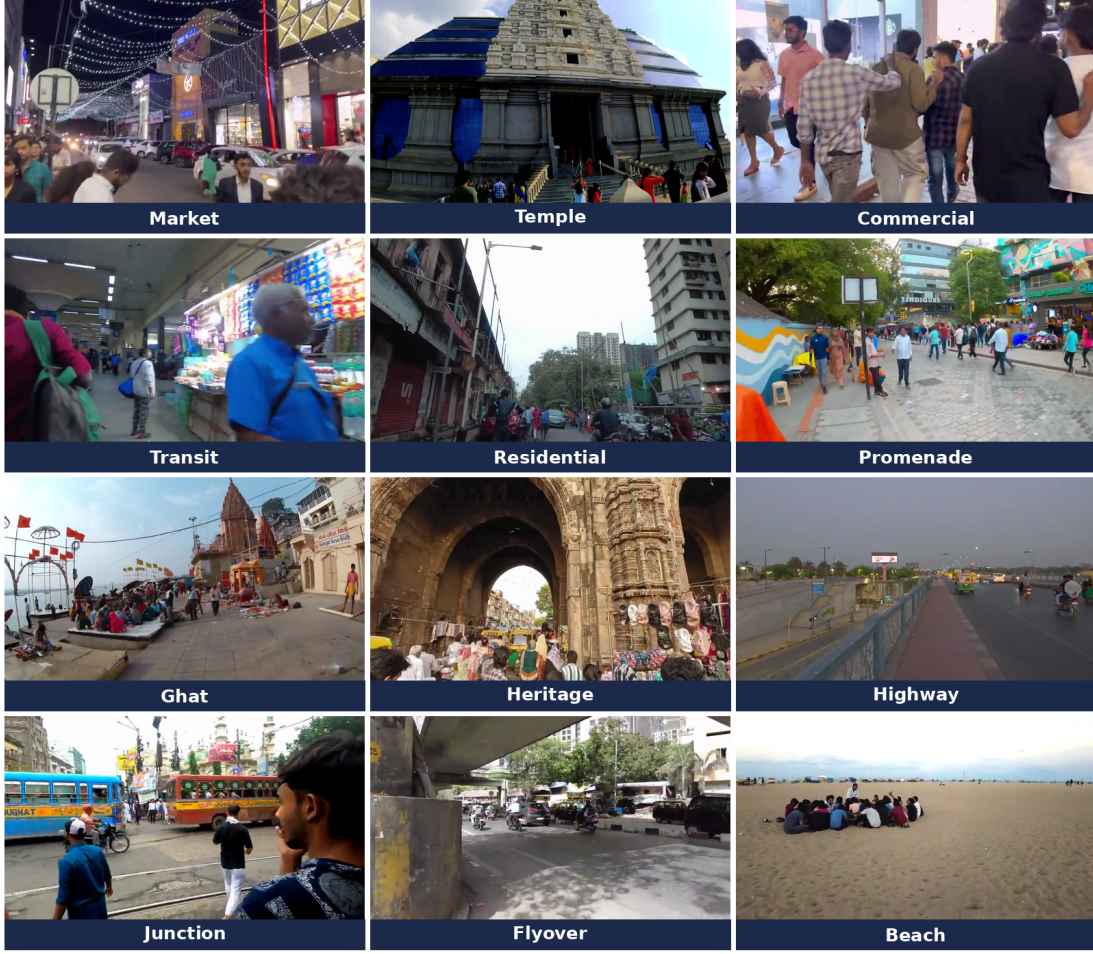


Figure 4: **DENSEWORLD 1.0 scene-type coverage.** Twelve representative scene types (market, temple, commercial, transit, residential, promenade, ghat, heritage, highway, junction, flyover, beach) across 22 Indian cities, shown from ground-level pedestrian viewpoints.

A deterministic structural map converts region geometry, visibility, temporal continuity, and relative motion into layout, agent, visibility, and interaction targets with reliability weights:

$$\Psi_{\text{struct}}(\mathcal{P}_n, \mathcal{T}_n) \mapsto \{(T_{n,k}, q_{n,k})\}_{k \in \{L, A, V, I\}}.$$

Factor-specific heads $\hat{T}_{n,k} = P_k(Z_{n,k})$ are trained using

$$\mathcal{L}_{\text{factor}} = \sum_{k \in \{L, A, I\}} \frac{\sum_{n=1}^N q_{n,k} \ell_k(\hat{T}_{n,k}, T_{n,k})}{\epsilon + \sum_{n=1}^N q_{n,k}},$$

while visibility is optimized separately through $\mathcal{L}_v$.

DINOv2-derived targets supervise training only and are never reused as evaluation labels. Checkpoints, association and interaction rules, thresholds, target coverage, and reliability audits are provided in Appendix .

This factorization is not only architectural: lightweight probes show the three channels emerge at distinct network depths (Figure 7).

Structured Factor Coordinates

For target token p , FactorJEPA forms the conditioned query

$$q_{n,p} = Q(f_\theta(M_c \odot x_n), \pi_p, M_t)$$

and predicts

$$c_{n,p} = \text{col}(c_{n,L,p}, c_{n,A,p}, c_{n,I,p}) \in \mathbb{R}^{r_L + r_A + r_I}.$$

We suppress p below and write $h_n = q_{n,p}$.

Layout. Slowly varying spatial support is encoded as

$$c_{n,L} = g_L(h_n), \quad \hat{T}_{n,L} = P_L(c_{n,L}).$$

Visibility-gated agents. For agent representation $o_n^{(i)}$,

$$s_n^{(i)} = g_A(h_n, o_n^{(i)}), \quad v_n^{(i)} = \sigma(g_V(h_n, o_n^{(i)})),$$

and

$$c_{n,A} = \frac{\sum_i v_n^{(i)} s_n^{(i)}}{\epsilon + \sum_i v_n^{(i)}}.$$

Table 1: **Frozen encoders collapse.** Ten frozen encoders on the DENSEWORLD motion probe ($n_{\text{test}} = 1,825$, $\pm 95\%$ BCa CI). Action top-1 (A) stays in a 37.5–44.4% band (19.5% majority), while adapted encoders reach 50.3–53.2%. M = motion-cos, T = taxonomy F1 ($\uparrow$ better). In each column the **best** value is shaded green and the **worst** red.

Frozen encoder	A (%)	M	T
V-JEPA 2.1 (2B)	44.4	0.009	0.793
V-JEPA 2.1 ViT-L	44.2	0.004	0.788
V-JEPA 1 ViT-H	40.5	0.007	0.702
LeJEPA ViT-L	40.1	0.014	0.740
V-JEPA 1 ViT-L	39.9	0.008	0.660
I-JEPA ViT-H	39.1	0.016	0.781
V-JEPA 2.0 (SSv2)	38.8	0.007	0.776
DINOv2	38.5	0.016	0.816
V-JEPA 2 ViT-L	37.9	0.013	0.778
I-JEPA ViT-G/16	37.5	0.019	0.787

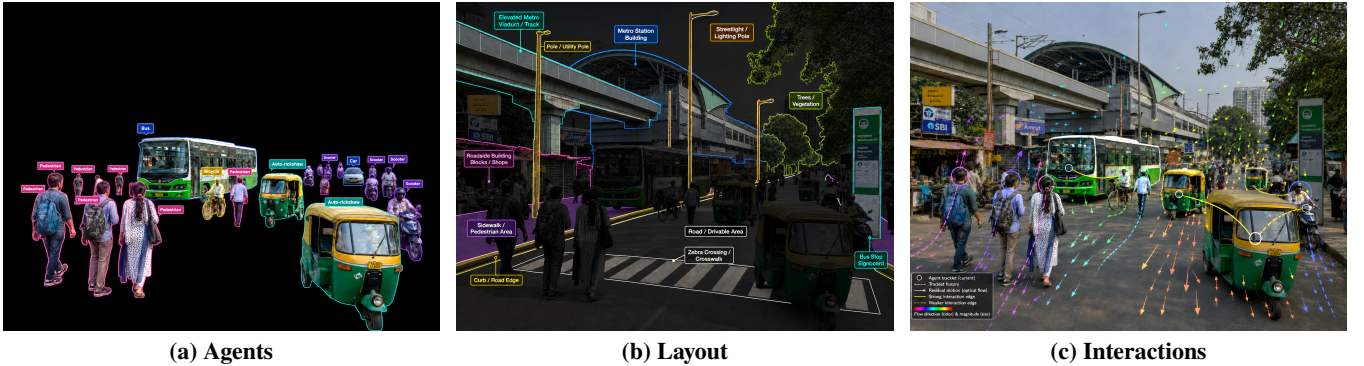


Figure 5: **Factorized decomposition of the same urban scene.** (a) Agents are isolated from scene context. (b) Layout highlights persistent spatial structure while retaining suppressed agent silhouettes. (c) Interactions show tracklets, residual motion, and sparse pairwise coupling; arrow direction and length encode motion direction and magnitude, while solid and dashed edges denote stronger and weaker interactions.

Table 2: **Protocol-matched V-JEPA adaptation baselines.** All methods retain the JEPA objective and train on the same raw clips; only the adaptation mechanism changes.

Method	Adaptation mechanism	Question tested
LoRA (Hu et al. 2022)	Adds low-rank updates, $\Delta W = (\alpha/r)BA$, to selected frozen projections.	Is parameter-efficient low-rank adaptation sufficient?
DoRA (Liu et al. 2024)	Separates weight magnitude from direction and applies low-rank updates to the directional component.	Does weight decomposition recover structure missed by LoRA?
Auto-RGN (Lee et al. 2023)	Selects transformer blocks using their relative gradient norms and updates only the selected subset.	Can gradient-guided selection localize the required adaptation?

The gate softly suppresses uncertain or occluded agents without removing them. Given visibility targets $T_{n,V}^{(i)}$ and reli-

bilities $q_{n,V}^{(i)}$,

$$\mathcal{L}_v = - \frac{\sum_{n,i} q_{n,V}^{(i)} [T_{n,V}^{(i)} \log v_n^{(i)} + (1 - T_{n,V}^{(i)}) \log(1 - v_n^{(i)})]}{\epsilon + \sum_{n,i} q_{n,V}^{(i)}}.$$

Sparse interactions. For each candidate pair $(i, j) \in \mathcal{E}_n$,

$$e_n^{(ij)} = g_I(h_n, o_n^{(i)}, o_n^{(j)}), \quad \bar{e}_n^{(ij)} = \frac{e_n^{(ij)}}{\epsilon + \|e_n^{(ij)}\|_2},$$

with soft interaction strength

$$w_n^{(ij)} = \sigma(g_W(h_n, o_n^{(i)}, o_n^{(j)})).$$

Let $M_n = \max\{1, |\mathcal{E}_n|\}$. Then

$$c_{n,I} = \frac{1}{M_n} \sum_{(i,j) \in \mathcal{E}_n} w_n^{(ij)} \bar{e}_n^{(ij)}, \quad c_{n,I} = 0 \text{ if } \mathcal{E}_n = \emptyset,$$

and localized interactions are encouraged through

$$\mathcal{L}_{\text{sparse}} = \frac{1}{N} \sum_{n=1}^N \frac{1}{M_n} \sum_{(i,j) \in \mathcal{E}_n} w_n^{(ij)}.$$

The fixed normalization prevents trivial rescaling of interaction weights and states.

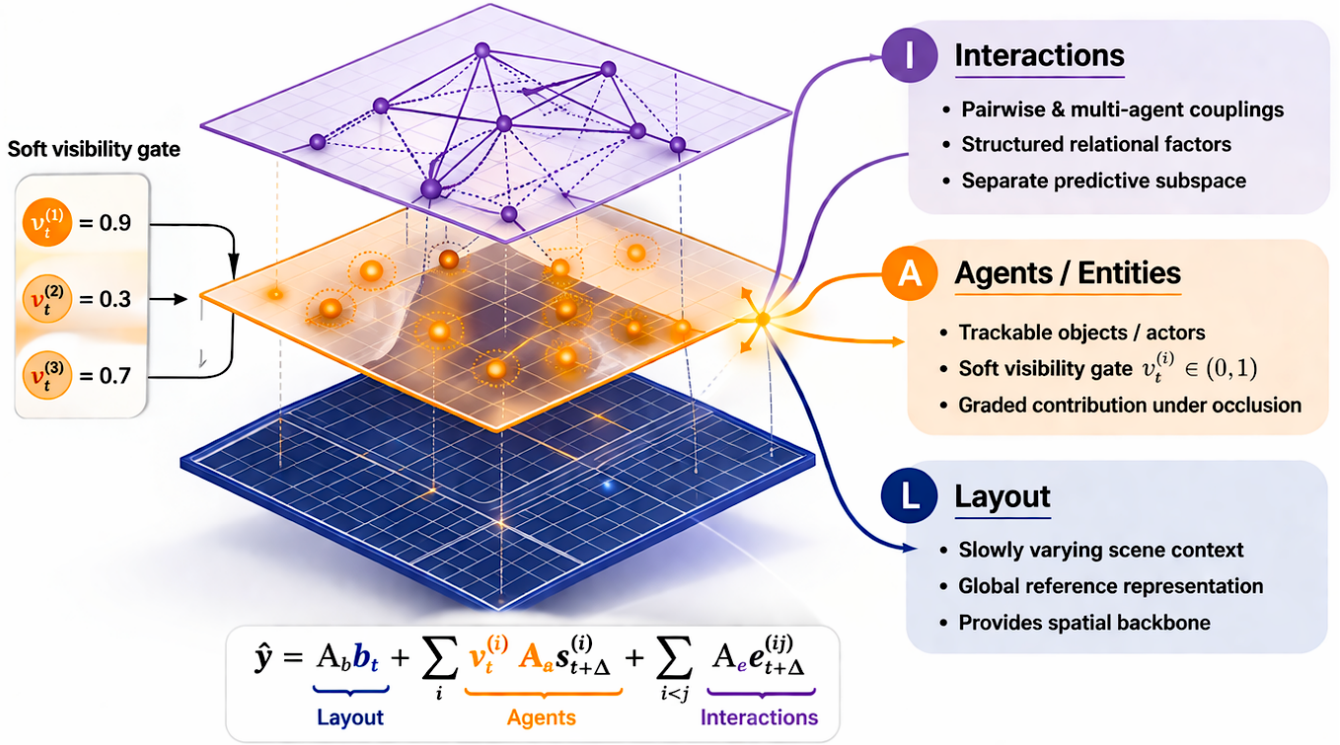


Figure 6: **Factorized prediction.** *FactorJEPA* composes the target embedding from *layout*, *agents*, and *interactions*, with a soft visibility gate applied only to entity terms.

Block-Structured Matrix Factorization

For each factor $k \in \{L, A, I\}$, we stack all target-token coordinates across the minibatch:

$$C_k = \begin{bmatrix} (c_{1,k,1})^\top \\ \vdots \\ (c_{1,k,m})^\top \\ \vdots \\ (c_{N,k,m})^\top \end{bmatrix} \in \mathbb{R}^{Nm \times r_k}.$$

The factor coordinates and learned synthesis dictionaries are concatenated as

$$C = [C_L \ C_A \ C_I], \quad A = [A_L \ A_A \ A_I], \quad A_k \in \mathbb{R}^{d \times r_k}.$$

FactorJEPA predicts the complete token-level target matrix through

$$\hat{Y} = \underbrace{[C_L \ C_A \ C_I]}_{\text{factor coordinates}} \underbrace{[A_L \ A_A \ A_I]^\top}_{\text{synthesis dictionaries}} = CA^\top$$

or, equivalently,

$$\hat{Y} = \underbrace{C_L A_L^\top}_{Y_L} + \underbrace{C_A A_A^\top}_{Y_A} + \underbrace{C_I A_I^\top}_{Y_I}.$$

Distinct pathways, factor supervision, and dictionaries make the decomposition architectural rather than post hoc.

For stacked future targets

$$Y^* = \text{col}(Y_{1,t+\Delta}^*, \dots, Y_{N,t+\Delta}^*) \in \mathbb{R}^{Nm \times d},$$

the JEPA objective is

$$\mathcal{L}_{\text{JEPA}} = \frac{1}{Nm} \left\| \hat{Y} - Y^* \right\|_F^2.$$

The corresponding m rows are unstacked per clip, preserving the complete future-token geometry.

The head-to-head advantage above holds up when FactorJEPA is placed against every adaptation family at once: the full scorecard (Figure 9) spans predictive, motion, semantic, and temporal diagnostics at both scales.

Semantic Anchoring and Channel Separation

The factorization $\hat{Y} = CA^\top$ is non-unique: for any invertible R ,

$$CA^\top = (CR)(AR^{-\top})^\top.$$

Thus, the JEPA objective alone permits rotations and cross-block mixing. Factor-specific heads

$$\hat{T}_k = C_k P_k^\top, \quad k \in \{L, A, I\},$$

together with $\mathcal{L}_{\text{factor}}$, anchor the blocks to layout, agents, and interactions. FactorJEPA therefore claims semantic block separation, not coordinate-level identifiability.

To suppress residual cross-channel shortcuts, let $\Gamma_{kk'}$ denote the empirical covariance between channel outputs Y_k

FactorJEPA diagnostic: factor information flow vs depth

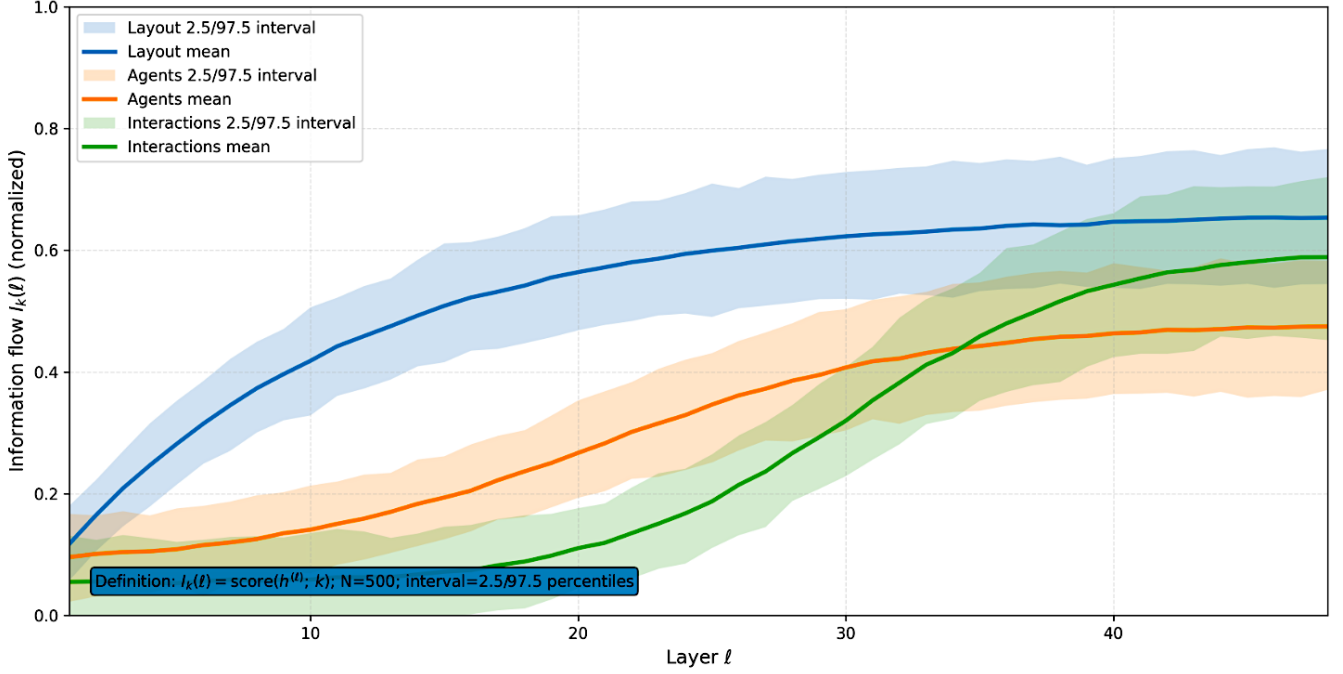


Figure 7: **Depth-wise factor realization.** Lightweight probes show a staged profile: *layout* emerges early, *agents* grow gradually, and *interactions* peak in deeper layers.

and $Y_{k'}$. We penalize linear and nonlinear leakage through

$$\mathcal{L}_{\text{sep}} = 2 \underbrace{\sum_{k < k'} \|\Gamma_{kk'}\|_F^2}_{\mathcal{L}_{\text{cov}}} + \beta_{\text{nl}} \underbrace{\sum_{k < k'} \frac{\text{tr}(\bar{K}_k \bar{K}_{k'})}{\|\bar{K}_k\|_F \|\bar{K}_{k'}\|_F + \epsilon}}_{\mathcal{L}_{\text{nl}}},$$

where $\bar{K}_k$ is the centered RBF Gram matrix for channel k . This suppresses cross-channel dependence while leaving within-channel variation unconstrained; it does not imply statistical or causal independence.

Leakage diagnostic. We freeze FactorJEPA and train fixed-capacity probes from channel Y_k to factor target $T_{k'}$. With held-out probe score $S_{k \rightarrow k'}$ and constant-predictor score $S_{0 \rightarrow k'}$, normalized leakage is

$$\text{Leak}(k \rightarrow k') = \frac{S_{k \rightarrow k'} - S_{0 \rightarrow k'}}{S_{k' \rightarrow k'} - S_{0 \rightarrow k'} + \epsilon}, \quad k \neq k'.$$

Effective separation requires strong diagonal predictability and low off-diagonal leakage. Full definitions and probe protocols are provided in Appendix .

Predictor Surgery

FactorJEPA is initialized from a pretrained V-JEPA model. The online encoder, momentum target encoder, target construction, and masking policy are retained. Only the monolithic predictor is replaced:

$$\Theta_{\text{pred}} \rightarrow \Theta_{\text{fact}} = \{\Theta_L, \Theta_A, \Theta_I, \Theta_V, \Theta_W, A\}.$$

Training is restricted to the factorized predictor and the top K encoder blocks:

$$\Theta_{\text{train}} = \Theta_{\text{fact}} \cup \Theta_{\text{top-}K}.$$

The lower encoder blocks and momentum target network follow the frozen or momentum-updated protocol of the underlying V-JEPA implementation.

The final objective is

$$\mathcal{L} = \mathcal{L}_{\text{JEPA}} + \lambda_{\text{sep}} \mathcal{L}_{\text{sep}} + \lambda_{\text{sparse}} \mathcal{L}_{\text{sparse}} + \lambda_v \mathcal{L}_v + \lambda_{\text{sup}} \mathcal{L}_{\text{factor}}.$$

Each term has a distinct role:

- $\mathcal{L}_{\text{JEPA}}$: future-latent prediction,
- $\mathcal{L}_{\text{sep}}$: cross-channel leakage suppression,
- $\mathcal{L}_{\text{sparse}}$: interaction locality,
- $\mathcal{L}_v$: visibility calibration,
- $\mathcal{L}_{\text{factor}}$: semantic anchoring.

In practice, surgery runs as a staged factor curriculum (Figure 11): a short head-only warmup precedes progressive unfreezing of the top- K encoder blocks, while the predictive emphasis shifts from *layout* to *agents* to *interactions*.

Depth-Wise Factor Realization

Although the factorization acts at the predictor, we test how factor information becomes linearly accessible across encoder depth. Let

FactorJEPA vs. the best competitor, across scale: separation in 95%-CI units
(bars sorted best-to-worst by the 115k panel; dashed line = the 1× separation threshold)

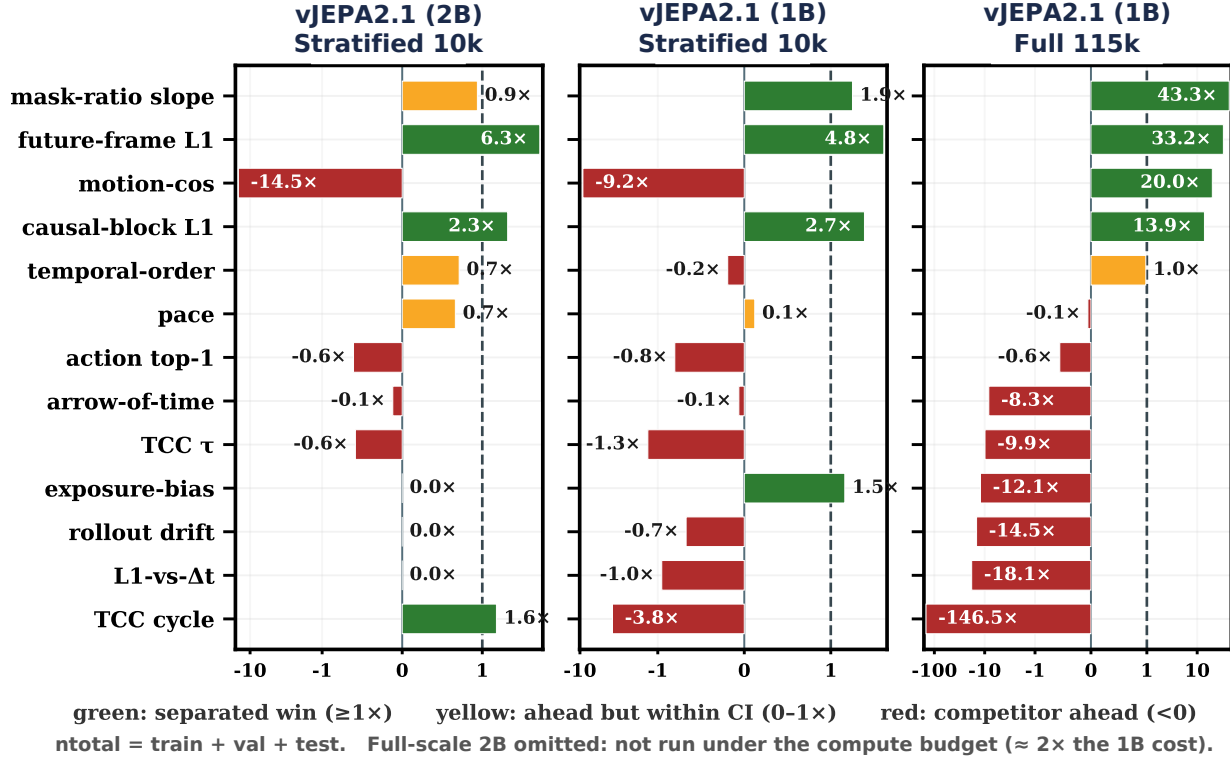


Figure 8: *FactorJEPA vs. the strongest competitor across scale and data regimes.* Bars report FactorJEPA’s advantage in units of the paired-difference 95% confidence interval; the dashed line marks statistical separation at 1×. Under matched stratified 10k protocol, FactorJEPA separates on Future-frame L1 and Causal L1 at both scales, and on Mask-ratio slope at 1B, while trailing on Motion cosine (revealing a consistent prediction–motion trade-off). With full 115k-clip training at 1B, it separates strongly on all four primary diagnostics, reaching 43.3× for Mask-ratio slope, 33.2× for Future-frame L1, 20.0× for Motion cosine, and 13.9× for Causal L1.

$$H^{(\ell)} = \begin{bmatrix} h_1^{(\ell)} & \dots & h_N^{(\ell)} \end{bmatrix}^\top \in \mathbb{R}^{N \times d_\ell}$$

denote the representations at layer ℓ . For each factor $k \in \{L, A, I\}$, we fit a regularized linear probe on the training split:

$$W_k^{(\ell)} = \arg \min_W \left\| H_{\text{train}}^{(\ell)} W - T_{k,\text{train}} \right\|_F^2 + \lambda_{\text{probe}} \|W\|_F^2$$

The held-out depth-wise factor score is

$$I_k(\ell) = \text{score}_k \left(H_{\text{test}}^{(\ell)} W_k^{(\ell)}, T_{k,\text{test}} \right).$$

Distinct onset and growth profiles indicate that layout, agent, and interaction information becomes accessible at different depths. This is a representation diagnostic, not a neuron-level or causal decomposition.

Experiments & Evaluation

We ask whether FactorJEPA improves V-JEPA’s *predictive structure*, rather than only its downstream semantics or linear accessibility. We evaluate four complementary diagnostics: **Future-frame L1**, **Causal L1**, **Mask-ratio slope**, and **Motion cosine**. Together, they probe future-latent fidelity, intervention sensitivity, robustness under partial observability, and accessible motion information, distinguishing improved forecasting from semantic alignment or easier linear readout alone.

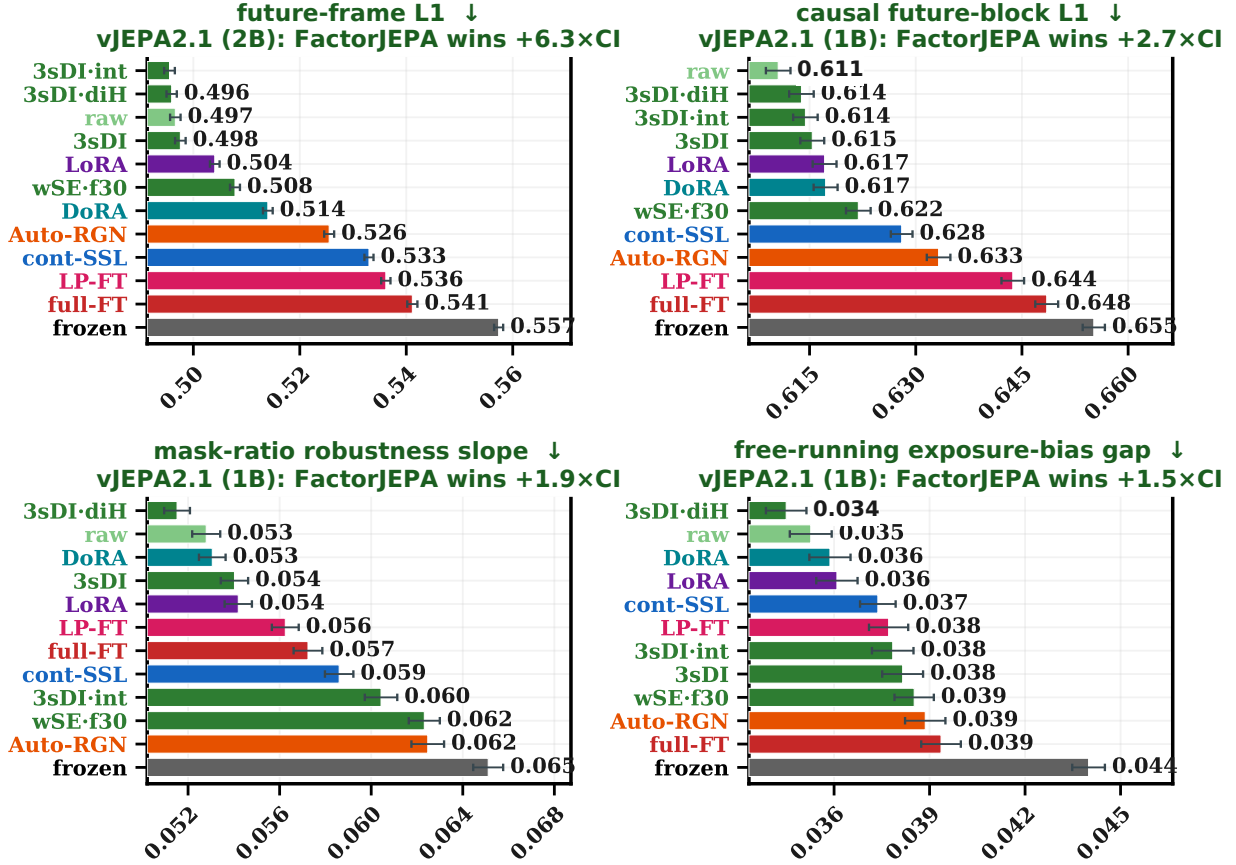
Evaluator Independence and Audit Split

To prevent agreement with the pseudo-label generator from being mistaken for improved world modeling, we strictly separate training-target construction from evaluation. DINOv2-derived targets are constructed only on the training split:

$$\mathcal{T}_{\text{train}} = \Psi_{\text{struct}}(\mathcal{F}_{\text{DINOv2}}(x)), \quad x \in \mathcal{D}_{\text{train}}.$$

Primary factor-specific evaluation uses a city-disjoint audit split $\mathcal{D}_{\text{audit}}$, whose agent masks, visibility states, interaction

FactorJEPA vs. every fine-tuning rival: the four predictive metrics it wins
(Stratified 10k · n_{test} = 1,825 · 95% BCa CI)



ENCODER KEY · short code = full name · ↓ = lower is better (all four are error metrics)
green shades = FactorJEPA / OURs · other colours = fine-tuning rivals
DI = interaction factor D_I · 3-stage = Layout → Agent → Interaction surgery

- | | |
|---|---|
| ■ 3sDI = FactorJEPA · 3-stage (L→A→interaction) | ■ DoRA = DoRA (weight-decomposed LoRA) |
| ■ 3sDI-diH = FactorJEPA · 3-stage (interaction-heavy) | ■ LP-FT = LP-FT (linear-probe then fine-tune) |
| ■ 3sDI-int = FactorJEPA · 3-stage + intervention | ■ LoRA = LoRA (low-rank adaptation) |
| ■ raw = FactorJEPA-RAW (no factor targets) | ■ cont-SSL = continual SSL (V-JEPA pretrain) |
| ■ wSE-f30 = FactorJEPA + WiSE-FT merge 30% | ■ frozen = frozen V-JEPA 2.1 encoder |
| ■ Auto-RGN = Auto-RGN gradient-norm surgery | ■ full-FT = full fine-tuning |

Figure 9: *Evaluation scorecard across 2B and 1B scales.* Frozen, conventionally adapted, parameter-efficient, and factorized V-JEPA 2.1 variants are compared across predictive, motion, semantic, and temporal diagnostics. Bars report performance with 95% BCa intervals where available.

pairs, and intervention regions are independently annotated:

$$\mathcal{D}_{\text{train}} \cap \mathcal{D}_{\text{audit}} = \emptyset, \quad \mathcal{T}_{\text{audit}} = \Psi_{\text{human}}(x).$$

Audit annotations are excluded from training, model selection, threshold selection, and hyperparameter tuning. City disjointness further limits scene-specific or geographic memorization. Future-frame L1 and Mask-ratio slope use the frozen V-JEPA target encoder; Motion cosine uses an independently frozen motion estimator; and RGB Agent F1 uses an external detector supplying neither FactorJEPA supervi-

sion nor DINOv2 preprocessing. Thus, no headline evaluator reuses FactorJEPA’s pseudo-label generator or training targets.

Protocol and Comparisons

We evaluate V-JEPA 2.1 at two scales: ViT-G with approximately 2B parameters and ViT-g with approximately 1B parameters. Both are evaluated under a matched stratified 10k-clip regime; ViT-g is additionally trained on the full

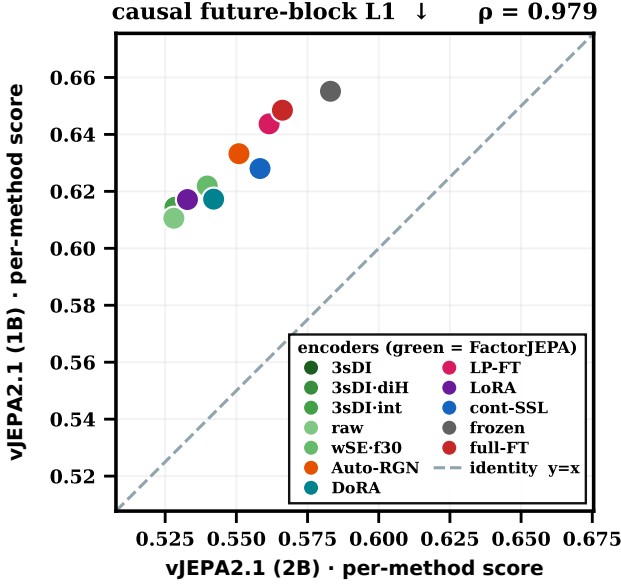


Figure 10: *Causal future-block rankings replicate across model scales.* Each point represents one adaptation method, comparing its causal future-block L1 score with the ViT-G 2B backbone on the horizontal axis and the ViT-g 1B backbone on the vertical axis; lower values are better. FactorJEPa variants are shown in green, and the dashed line denotes equal scores across scales. The strong Spearman correlation ($\rho = 0.979$) shows that the relative method ordering is highly preserved when scaling from 2B to 1B.

115k-clip corpus. Full-scale ViT-G training is omitted under the declared compute budget because it requires approximately twice the 1B cost.

Comparisons span the frozen backbone, full and parameter-efficient fine-tuning, continual SSL, LP-FT, LoRA, DoRA, Auto-RGN, WiSE-FT, **FactorJEPa-RAW**, and full **FactorJEPa**. FactorJEPa-RAW retains the factorized predictor but removes factor-target supervision, yielding

$$\underbrace{\{\text{LoRA, DoRA, Auto-RGN}\}}_{\text{generic adaptation}} \rightarrow \underbrace{\text{FactorJEPa-RAW}}_{\text{factorized architecture}} \rightarrow \underbrace{\text{FactorJEPa}}_{\text{architecture + factor curriculum}}$$

This attribution chain separates gains from generic adaptation, predictor factorization, and structured supervision. The experiments therefore test whether explicit **layout-agent-interaction structure** improves future prediction beyond a matched monolithic JEPa predictor.

Within each regime, all methods use identical clips, masking policies, optimization steps, evaluation splits, and metric implementations. We report per-encoder test results with paired 95% BCa bootstrap confidence intervals.

Four Diagnostics of Predictive Structure

Future-frame L1 ($\downarrow$). Normalized L1 distance between predicted and target future embeddings; lower values indicate *more accurate future-latent prediction*.

Causal L1 ($\downarrow$). For each independently annotated intervention region a , we apply the same controlled edit $\mathcal{I}_a$ and compare its effect on predicted and target future latents:

$$\hat{\Delta}_a = \hat{Y}(\mathcal{I}_a(x)) - \hat{Y}(x), \quad \Delta_a^* = \text{sg}[\bar{f}_{\bar{\theta}}(\mathcal{I}_a(x)) - \bar{f}_{\bar{\theta}}(x)].$$

$$\text{CausalL1} = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \frac{\|\hat{\Delta}_a - \Delta_a^*\|_1}{\epsilon + \|\Delta_a^*\|_1}.$$

Intervention regions and types come from independent audit annotations, not DINOv2 predictions. The metric evaluates consistency with intervention-induced changes, not causal identification.

Mask-ratio slope ($\downarrow$). Increase in prediction error as visual evidence is progressively masked; a smaller slope indicates *greater robustness under partial observability*.

Motion cosine ($\uparrow$). Cosine alignment between linearly decoded and target motion descriptors; higher values indicate *more linearly accessible motion information*.

Together, these diagnostics distinguish improved future modeling from gains limited to semantic decoding, a single masking level, or convenient linear representation.

Performance Across Scale and Data Regimes

Figures 8 and 9 summarize the primary evidence. Under the matched stratified 10k protocol, FactorJEPa separates from the strongest competitor on Future-frame L1 ($6.3 \times / 4.8 \times$ CI widths) and Causal L1 ($2.3 \times / 2.7 \times$) at 2B/1B. Mask-ratio slope remains within the confidence interval at 2B ($0.9 \times$) but separates at 1B ($1.9 \times$). Motion cosine instead favors generic fine-tuning ($-14.5 \times / -9.2 \times$), exposing a limited-data prediction-motion trade-off.

Full 115k-clip training changes this profile decisively. At 1B, FactorJEPa separates on all four primary diagnostics, reaching $43.3 \times$ for Mask-ratio slope, $33.2 \times$ for Future-frame L1, $20.0 \times$ for Motion cosine, and $13.9 \times$ for Causal L1 (Figure 8). These values denote separation in paired confidence-interval units, not multiplicative performance gains.

Figure 9 further shows that the gains are *structured rather than universal*. FactorJEPa leads on predictive fidelity, intervention sensitivity, and robustness-oriented diagnostics, while generic adaptation remains competitive on several frame-timing and temporal-coherence measures. The improvement therefore targets the information required to forecast crowded, partially observed scenes rather than uniformly shifting unrelated metrics.

Prediction-Motion Trade-off

Under stratified 10k training, FactorJEPa improves future prediction, intervention consistency, and masking robustness while conceding maximally linear motion readout to conventional fine-tuning. This suggests that, with limited data, the factorized objective prioritizes motion information useful for structured forecasting rather than motion that is easiest to recover linearly.

Crucially, full 115k training reverses this deficit: Motion cosine becomes a strongly separated win while the predictive gains increase further. The trade-off is therefore *data-*

Surgery · 86,831 clips × 1 ep × BS=32 × LR=5.0e-05 (2,713 steps)

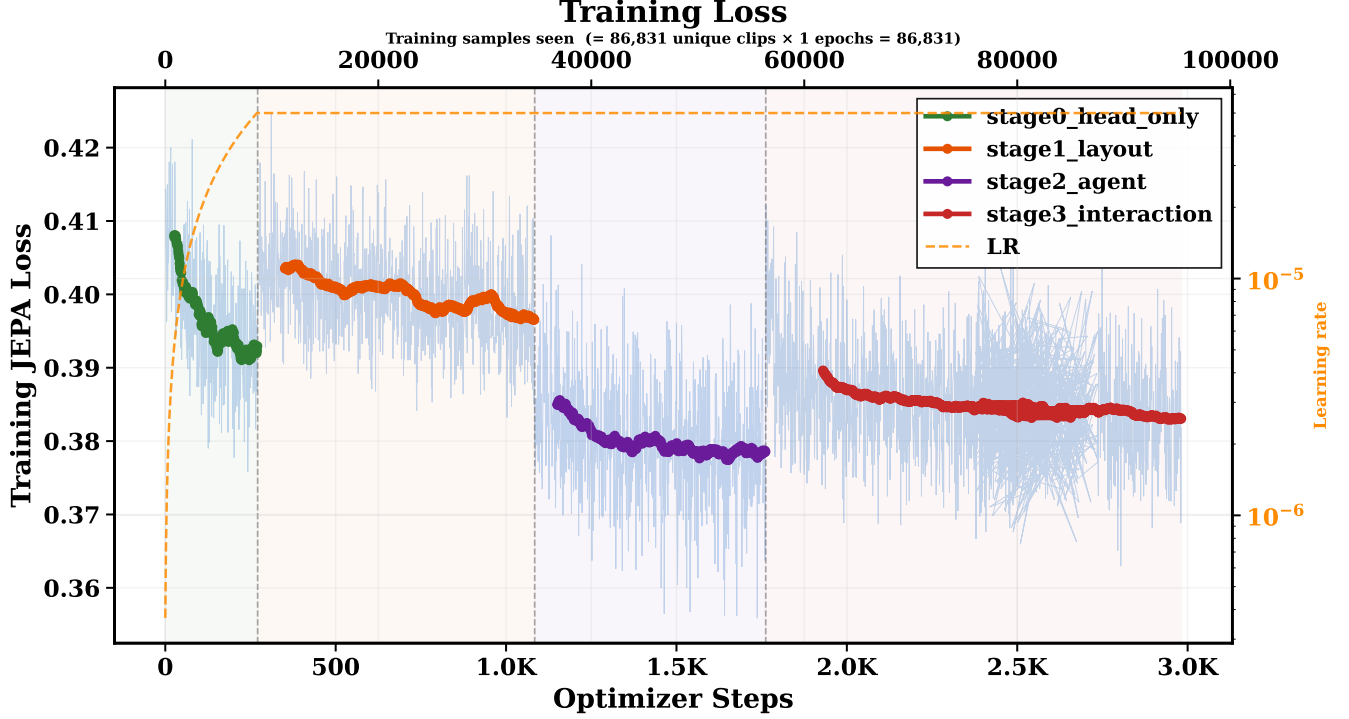


Figure 11: *Staged factor-curriculum surgery on the full corpus*. Training JEPa loss for FactorJEPa predictor surgery on the full DENSEWORLD corpus ($\sim 115k$ clips; ViT-g 1B backbone, batch 32, learning rate 5×10^{-5}). Surgery proceeds in four shaded phases (dashed boundaries): a *head-only* warmup, then progressive unfreezing of the top encoder blocks while the predictive emphasis cycles through *layout*, *agents*, and *interactions*. The loss falls from ≈ 0.41 to ≈ 0.38 ; each transition briefly raises the loss as the target distribution shifts, after which training re-descends, with the interaction stage settling at a slightly higher floor consistent with its harder prediction target. The learning-rate schedule is dashed (right axis).

and optimization-dependent, not an intrinsic limitation of layout–agent–interaction factorization. With sufficient interaction coverage, FactorJEPa preserves both structured prediction and accessible motion.

Cross-Scale Replication

We test whether adaptation rankings survive the transition from the 2B backbone to the approximately half-cost 1B backbone. For diagnostic k ,

$$\rho_k = \text{Spearman}(s_k^{2B}, s_k^{1B}),$$

where s_k^{2B} and s_k^{1B} contain per-method scores at the two scales.

As Figure 10 illustrates, Causal L1 yields $\rho = 0.979$, Motion cosine $\rho = 0.952$, Future-frame L1 $\rho = 0.938$, and Mask-ratio slope $\rho = 0.895$. These are the four strongest cross-scale correlations. Twelve of fifteen diagnostics retain broadly consistent rankings, while Temporal-order, Teacher-free gap, and Rollout drift do not transfer reliably.

The result shows that the relative behavior of adaptation strategies is largely backbone-invariant. Both FactorJEPa’s predictive gains and its limited-data motion trade-off reflect stable method ordering rather than artifacts of one model

size. The high correlations also establish the 1B backbone as a faithful lower-cost proxy for screening adaptation choices before full-scale 2B training.

Conclusion

We introduced **DENSEWORLD**, a 1,000-hour benchmark from 22 cities for world modeling under dense traffic, heterogeneity, occlusion, and partial observability. **FactorJEPa** decomposes future prediction into visibility-aware *layout*, *agent*, and *interaction* channels, yielding (i) *lower future-latent error*, (ii) *stronger intervention sensitivity*, and (iii) *greater robustness to missing evidence* across 2B and 1B backbones, with stable cross-scale rankings ($\rho = 0.895$ – 0.978). The results expose a consistent trade-off with linearly accessible motion. A Cosmos-initialized decoder further renders predicted futures, with an oracle control isolating forecasting error from decoder limitations. FactorJEPa therefore moves JEPa world models from monolithic latent prediction toward structured, interpretable forecasting of complex urban dynamics.

References

- Asaithambi, G.; Kanagaraj, V.; Srinivasan, K. K.; and Sivanandan, R. 2012. Characteristics of Mixed Traffic on Urban Arterials with Significant Volumes of Motorized Two-Wheelers: Role of Composition, Intra-class Variability, and Lack of Lane Discipline. *Transportation Research Record: Journal of the Transportation Research Board*, 2317(1): 51–59.
- Caesar, H.; Bankiti, V.; Lang, A. H.; Vora, S.; Liong, V. E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; and Beijbom, O. 2020. nuScenes: A Multimodal Dataset for Autonomous Driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 11621–11631.
- Dokania, S.; Hafez, A. H. A.; Subramanian, A.; Chandraker, M.; and Jawahar, C. V. 2023. IDD-3D: Indian Driving Dataset for 3D Unstructured Road Scenes. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 4482–4491.
- Ellis, P.; and Roberts, M. 2016. *Leveraging Urbanization in South Asia: Managing Spatial Transformation for Prosperity and Livability*. South Asia Development Matters. Washington, DC: World Bank. ISBN 978-1-4648-0662-9.
- Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*.
- Khan, S. I.; and Maini, P. 1999. Modeling Heterogeneous Traffic Flow. *Transportation Research Record: Journal of the Transportation Research Board*, 1678(1): 234–241.
- Lee, Y.; Chen, A. S.; Tajwar, F.; Kumar, A.; Yao, H.; Liang, P.; and Finn, C. 2023. Surgical Fine-Tuning Improves Adaptation to Distribution Shifts. In *International Conference on Learning Representations*.
- Liu, S.-Y.; Wang, C.-Y.; Yin, H.; Molchanov, P.; Wang, Y.-C. F.; Cheng, K.-T.; and Chen, M.-H. 2024. DoRA: Weight-Decomposed Low-Rank Adaptation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, 32100–32121. PMLR.
- Mahendra, A.; King, R.; Du, J.; Dasgupta, A.; Beard, V. A.; Kallergis, A.; and Schalch, K. 2021. Seven Transformations for More Equitable and Sustainable Cities. Technical report, World Resources Institute, Washington, DC.
- NVIDIA. 2025. Cosmos World Foundation Model Platform for Physical AI. *arXiv preprint arXiv:2501.03575*.
- Oquab, M.; Darcet, T.; Moutakanni, T.; Vo, H. V.; Szafraniec, M.; Khalidov, V.; Fernandez, P.; Haziza, D.; Massa, F.; El-Nouby, A.; Assran, M.; Ballas, N.; Galuba, W.; Howes, R.; Huang, P.-Y.; Li, S.-W.; Misra, I.; Rabbat, M.; Sharma, V.; Synnaeve, G.; Xu, H.; Jégou, H.; Mairal, J.; Labatut, P.; Joulin, A.; and Bojanowski, P. 2024. DINOv2: Learning Robust Visual Features without Supervision. *Transactions on Machine Learning Research*.
- Papathanasopoulou, V.; and Antoniou, C. 2018. Flexible Car-Following Models for Mixed Traffic and Weak Lane-Discipline Conditions. *European Transport Research Review*, 10(2): 62.
- Varma, G.; Subramanian, A.; Namboodiri, A.; Chandraker, M.; and Jawahar, C. V. 2019. IDD: A Dataset for Exploring Problems of Autonomous Navigation in Unconstrained Environments. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, 1743–1751.
- Yu, F.; Chen, H.; Wang, X.; Xian, W.; Chen, Y.; Liu, F.; Madhavan, V.; and Darrell, T. 2020. BDD100K: A Diverse Driving Dataset for Heterogeneous Multitask Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2636–2645.

Main-Paper Figures and Tables, Enlarged

For legibility, this section reproduces each main-paper figure and the ablation table at full width, in the order they appear in the main text, with a brief note linking them.

FactorJEPA meets this by decomposing the future into three predictive channels, layout, visibility-gated agents, and sparse interactions, recomposed in the JEPA embedding to discourage cross-factor shortcuts.

These three views are constructed automatically by a fixed detection-and-segmentation pipeline, so the factor supervision is reproducible and scales to the full corpus without manual labels.

Limitations

FactorJEPA demonstrates that structured predictive channels can improve world modeling in dense, heterogeneous, and partially observed urban scenes. The present evidence nevertheless leaves several important questions open, particularly around interaction grounding, semantic factorization, long-horizon evaluation, geographic transfer, and full-scale training.

(i) Interaction grounding remains the principal open challenge. Our interaction targets combine frozen DINOv2 regions, temporal association, relative motion, visibility, proximity, and reliability weighting. This design provides scalable supervision across 1,000 hours of video and captures a tractable subset of the interaction cues that matter for prediction. However, urban interactions are not directly segmentable visual objects: they are temporally extended relations shaped by anticipation, yielding, hesitation, shared right of way, informal signaling, and partially observed actors. Nearby agents may move independently, whereas distant or occluded agents may still influence the future. The current pathway therefore emphasizes sparse pairwise coupling and does not yet fully represent **(i) group-level motion**, **(ii) multi-agent negotiation**, or **(iii) temporally persistent interaction events**. The independently annotated audit split ensures that the training-time interaction generator is not reused as the headline evaluator, while reliability weighting reduces the influence of uncertain targets. Even so, richer event-level annotation, trajectory-aware grounding, and higher-order relational representations could provide more complete supervision. Accordingly, FactorJEPA should be interpreted as learning **interaction-sensitive predictive channels**, while richer semantic, intentional, and causal interaction modeling remains open.

(ii) Factor semantics are anchored at the channel level. Layout, agent, visibility, and interaction targets are produced by a fixed DINOv2-based pipeline rather than exhaustive human annotation. This enables supervision at dataset scale, but the resulting targets may still reflect missed small agents, fragmented tracks, uncertain boundaries, or reduced reliability under severe occlusion, blur, poor illumination, and camera motion. Some distinctions are also inherently contextual: parked vehicles, temporary barriers, vendors, crowds, and encroachments may behave as persistent layout in one scene and dynamic entities in another. Factor-specific pathways, heads,

and separation losses provide **semantic anchoring and functional channel separation**, but do not imply coordinate-level identifiability, complete statistical independence, or a unique decomposition of the future latent. Alternative teachers, taxonomies, or factor ranks may yield different yet comparably predictive partitions. Multi-teacher agreement, teacher-free factor discovery, and intervention-based equivalence tests are therefore natural extensions.

(iii) The evaluation targets predictive structure rather than complete causal understanding. The four primary diagnostics have deliberately bounded interpretations. Future-frame L1 measures fidelity in the frozen target-encoder space; Mask-ratio slope measures robustness under controlled evidence removal; Motion cosine measures linearly accessible motion rather than all motion information encoded by the model; and Causal L1 evaluates consistency between predicted and target-encoder changes under independently specified interventions. Causal L1 therefore measures **intervention sensitivity**, not causal identification or unrestricted counterfactual reasoning. The experiments also emphasize short-horizon prediction from prerecorded video. Longer rollouts may accumulate identity, layout, and interaction errors, while action-conditioned forecasting, closed-loop planning, active perception, and online adaptation remain untested. The current results thus establish FactorJEPA as a **structured predictive representation**; extending it to long-horizon and closed-loop world modeling remains future work.

(iv) Geographic breadth does not imply universal coverage. DENSEWORLD spans 22 Indian cities, three capture modes, and substantial variation in density, infrastructure, illumination, weather, road structure, and traffic composition. It should nevertheless be viewed as **one large-scale realization** of populous, crowded, and chaotic Global South urban environments rather than an exhaustive characterization of the Global South. Mobility systems, infrastructure, road conventions, climate, and social negotiation differ across South Asia, Africa, Latin America, Southeast Asia, and the Middle East. The corpus also emphasizes outdoor urban public spaces; rural roads, indoor crowds, industrial environments, disaster settings, and other interaction regimes remain outside its present scope. Cross-region transfer, geographically held-out evaluation, and broader collection are needed to test how far the learned factorization generalizes.

(v) Full-scale training and visual decoding remain incomplete. Full 115k-clip training is reported for the 1B backbone, whereas the 2B model is evaluated under the matched stratified 10k regime. The strong cross-scale rank correlations indicate stable relative method behavior, but they do not replace a full-data 2B experiment. It therefore remains open whether larger-scale training further strengthens the predictive channels, changes their depth-wise allocation, or modifies the observed prediction–motion trade-off. The Cosmos-initialized latent-to-RGB decoder also introduces a separate rendering stage. Oracle controls isolate forecast-induced error from decoder reconstruction error, but rendering may still exhibit blur, deterministic averaging, missed fine-grained

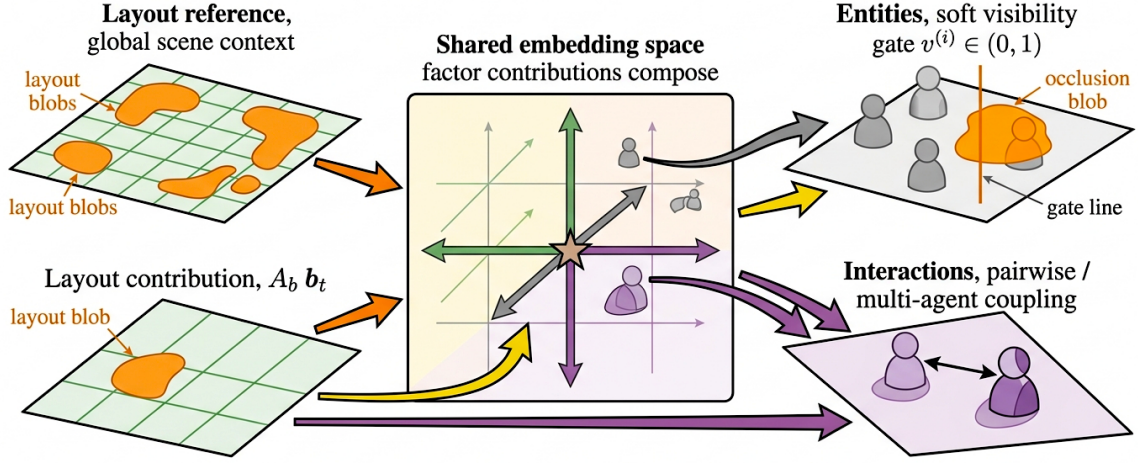


Figure 12: **Factorized predictive channels.** Context is split into *layout*, visibility-gated *agent*, and sparse *interaction* coordinates, then recomposed in the future JEPA embedding, limiting cross-factor leakage.



Figure 13: **Automatic factor-target construction.** Per clip, the pipeline derives *layout* D_L , *agent* D_A , and *interaction* D_I supervision by masking foreground/background and linking agent tubes.

agents, or uncertainty collapsed into a single future. Visual plausibility should therefore be interpreted alongside latent-space and oracle-controlled evaluation rather than as standalone evidence of predictive correctness.

These limitations motivate three immediate directions: (i) **richer interaction grounding and event-level supervision**, (ii) **higher-order and temporally persistent relational modeling**, and (iii) **broadier geographic, long-horizon, and full-scale validation**.

Appendix

This appendix provides the complete evidence and implementation record underlying the main paper. Its organization follows the scientific progression of the study. Appendices A–C establish the provenance, distinctiveness, and reproducibility of DENSEWORLD, its factor targets, and the executed training protocols. Appendices D–F then examine the central methodological claims: which components produce the observed gains, whether the evaluation is statistically reliable, and whether the learned layout, agent, and interaction channels are functionally distinct. Appendix G evaluates the

temporal and downstream consequences of the prediction–motion trade-off. Finally, Appendix H documents the latent-to-RGB decoder, separates forecast error from reconstruction limitations, and presents quantitative and qualitative future predictions.

The appendix distinguishes primary confirmatory evidence from diagnostic and robustness analyses. Unless otherwise stated, comparisons use identical evaluation clips, paired resampling units, and the same independently defined audit signals. Factor separation is interpreted operationally at the level of predictive channels rather than as coordinate-level identifiability or complete statistical independence. Intervention-based evaluation measures consistency with controlled edits, while cross-scale replication refers specifically to stability between the 1B and 2B V-JEPA 2.1 backbones.

Appendix index.

- **Appendix A: DENSEWORLD Construction, Splits, and Responsible Use**

- **A.1 Acquisition protocol and capture modes**

Shared collection specification; geographic and environmental sampling; non-scripted scene activity; and drive-through, walk-through, and aerial capture.

- **A.2 Source processing, clip construction, and provenance**

FFmpeg decoding; adaptive shot-boundary detection; model-ready clip construction; quality filtering; and the retained city–session–source–shot–clip provenance hierarchy.

- **A.3 Partitioning and DINOv2 target provenance**

Source-grouped 80/10/10 train–validation–test allocation; partition independence; permitted information flow; and DINOv2-derived training, validation, and teacher-relative test targets. No manually annotated semantic labels are used.

- **A.4 Corpus composition and coverage**

City and capture-mode distributions; environmental and temporal coverage; scene families; traffic composition; and the automated dynamic-agent taxonomy, including locally distinctive agents.

- **A.5 Privacy processing, responsible use, and research artifacts**

Automated face and registration-plate detection; temporal association and blurring; redaction quality control; governed research access; supporting code and aggregate statistics; intended and prohibited uses; and empirical scope and extensions.

- **Appendix B: Five-Axis Validation of the DENSEWORLD Regime**

- **B.1 Shared measurement protocol and automated taxonomy**

Common preprocessing, temporal sampling, valid-image regions, DINOv2-based extraction, dynamic-agent taxonomy, and source-level aggregation across DENSEWORLD, BDD100K, and nuScenes.

- **B.2 Definition of the five regime axes**

Formal definitions of agent count density, agent occupancy, occlusion pressure, interaction pressure, and agent heterogeneity, including normalization, aggregation units, and edge-case handling.

- **B.3 Matched cross-dataset comparison**

Road-level scene matching by capture configuration, scene type, illumination, weather, image resolution, field of view, and clip duration. Walk-through and aerial DENSEWORLD clips are analyzed separately from the primary driving-benchmark comparison.

- **B.4 Comparative results, uncertainty, and sensitivity**

Five-axis estimates, scene-stratified effects, source-video-clustered confidence intervals, and hierarchical resampling over cities or sequences. Sensitivity is evaluated over detection threshold, minimum agent size, valid-image normalization, and interaction parameters. Results are interpreted as evidence for an Indian urban realization of the broader DENSEWORLD regime, not as an exhaustive characterization of Global-South mobility.

- **Appendix C: Segmentation and Training**

- **C.1 Frozen Teacher and Factor-Target Construction**

DINOv2 checkpoint, preprocessing, feature extraction, region and mask construction, confidence filtering, temporal association, and the exact definitions of the layout, agent, visibility, and interaction targets T_L, T_A, T_V, T_I . This subsection also specifies target dimensions, interaction-pair construction, reliability weights, exclusion rules, and retained target coverage.

- **C.2 FactorJEPA Architecture and Prediction Protocol**

V-JEPA 2.1 backbone scales, frozen and trainable components, predictor depth, factor projections, subspace ranks, visibility gating, masking policy, context duration, prediction horizon, and factor-to-predictor information flow.

- **C.3 Optimization, Model Selection, and Executed Algorithm**

Objective terms, loss weights, optimizer, learning-rate schedule, momentum-target update, batch size, gradient clipping, training steps, random seeds, validation criteria, checkpoint selection, and end-to-end training pseudocode.

- **C.4 Baseline Matching and Attribution Controls**

Full fine-tuning, LoRA, DoRA, Auto-RGN, FactorJEPA-RAW, and FactorJEPA configurations; trainable-parameter matching; shared data, masks, objectives, training steps, and evaluation protocol; and the controls separating generic adaptation, factorized architecture, and factor-target supervision.

- **C.5 Evaluation Provenance and Resource Accounting**

Separation of training targets, validation signals, and frozen test evaluators; software and hardware configu-

ration; trainable and total parameters; GPU-hours; peak memory; throughput; inference latency; and the reproducibility record for every reported experiment.

- **Appendix D: Attribution and Component Ablations**

- **D.1 Attribution design and experimental contract**
Defines the architectural and supervisory interventions, the parameter-matching protocol, the executed objectives, and the statistical estimands used throughout the appendix.
- **D.2 Architecture $\times$ structured-supervision factorial**
Compares matched monolithic and factorized predictors with and without DINOv2-derived structured supervision, isolating architecture, supervision, and their interaction.
- **D.3 Factor pathways and objective ablations**
Tests the layout–agent target loss, visibility pathway, interaction pathway, sparsity regularization, and separation objective through controlled one-factor removals.
- **D.4 Falsification controls and sensitivity**
Evaluates temporally misaligned and clip-shuffled teacher targets, reliability weighting, factor ranks, and loss coefficients to distinguish structured information from generic regularization.
- **D.5 Attribution synthesis**
Consolidates the factorial contrasts and component ablations, reporting which mechanisms account for each prediction, visibility, interaction, motion, and RGB result.

- **Appendix E: Metrics, Clustered Inference, and Robustness**

- **E.1 Evaluation contract and metric registry**
Defines every reported diagnostic by formula, direction, evaluation unit, aggregation rule, frozen evaluator, provenance, and primary or exploratory status.
- **E.2 Primary predictive diagnostics**
Specifies Future-frame MSE, intervention-consistency L1, Mask-ratio slope, and Motion cosine, including token aggregation, normalization, probe training, and horizon or masking support.
- **E.3 Secondary semantic and temporal diagnostics**
Defines action, taxonomy, rollout, temporal-order, arrow-of-time, playback-pace, temporal-correspondence, and exposure-bias diagnostics under a common aggregation protocol.
- **E.4 Paired clustered inference and multiplicity**
Defines the city–source-video estimand, hierarchical paired bootstrap, treatment of training seeds, mixed-effects sensitivity, and multiple-comparison control.
- **E.5 Metric sensitivity and complete numerical results**
Reports horizon and masking curves, intervention-normalization sensitivity, cross-scale ranking robustness, leave-one-method and leave-one-family analyses, and the complete ViT-G and ViT-g numerical scorecards.

- **Appendix H: Latent-to-RGB Decoding and Qualitative Results**

- **H.1 Cosmos checkpoint and decoder components**
Exact initialization, frozen modules, trainable modules, and adaptation configuration.
- **H.2 Latent transport architecture**
Token projection, spatial reshaping, positional encoding, normalization, and transport dimensions.
- **H.3 Decoder training objective**
Pixel, perceptual, structural, motion, and regularization losses.
- **H.4 Decoder-neutrality controls**
Target-latent-only, balanced-prediction, and method-specific decoder protocols.
- **H.5 Oracle, forecast-gap, and end-to-end evaluation**
Operational decomposition of reconstruction and predicted-latent errors.
- **H.6 Quantitative RGB results**
PSNR, SSIM, LPIPS, Flow EPE, Agent F1, dynamic-region metrics, and horizon-wise results.
- **H.7 Density- and occlusion-stratified RGB evaluation**
Rendering quality across low-density, high-density, and heavily occluded conditions.
- **H.8 Factor-sensitive visual diagnostics**
Layout-, agent-, and interaction-removed decoding and influence maps.
- **H.9 Qualitative future-prediction gallery**
Context, target, oracle, V-JEPA, FactorJEPA-RAW, and FactorJEPA comparisons.
- **H.10 Failure taxonomy**
Layout drift, missed agents, identity errors, incorrect interactions, motion errors, deterministic averaging, and decoder hallucination.
- **H.11 Additional qualitative examples**
Examples across cities, scene types, viewpoints, weather, illumination, density, and prediction horizons.

DENSEWORLD Construction, Splits, and Responsible Use

DENSEWORLD comprises approximately **1,000 hours** of drive-through, walk-through, and aerial video acquired across **22 Indian cities**. It targets predictive modeling under high agent density, heterogeneous traffic, persistent occlusion, soft spatial boundaries, and frequent local interaction. DENSEWORLD contains *no manually annotated semantic labels*; all layout, agent, visibility, and interaction targets are produced automatically using a fixed DINOv2-based pipeline.

Acquisition Protocol and Capture Modes

DENSEWORLD was acquired through two professional video-collection organizations following a **shared acquisition specification**. The specification defined coverage targets

over geographic sites, scene types, times of day, weather, visibility, road structure, and infrastructure conditions. It controlled *where and how* videos were acquired without scripting traffic, pedestrian, or animal behavior.

The collection uses three capture modes. **Drive-through** recordings emphasize road-level ego-motion, mixed traffic, and near-field interactions. **Walk-through** recordings capture pedestrian-scale navigation, shared spaces, markets, commercial corridors, and transit areas. **Aerial** recordings provide broader context for crowd flow, junction organization, traffic structure, and interaction topology. City, collection session, capture mode, and source timestamps are retained as acquisition metadata.

Source Processing, Clip Construction, and Provenance

Source videos are decoded using FFmpeg¹ while retaining frame timestamps and source identifiers. Shot boundaries are detected using the PySceneDetect AdaptiveDetector², which normalizes adjacent-frame changes in hue, saturation, and luminance by a rolling temporal average. This reduces false boundaries caused by rapid ego-motion and camera shake.

We use a fixed corpus-wide configuration: `adaptive_threshold= 3.0`, `min_content_val= 15.0`, `window_width= 2`, and `min_scene_len= 15` frames. For a source video $v = \{x_t^{(v)}\}_{t=1}^{T_v}$, the detector returns ordered boundaries

$$\mathcal{B}_v = \left\{ 1 = b_0^{(v)} < b_1^{(v)} < \dots < b_{J_v}^{(v)} = T_v \right\},$$

which partition the source into candidate visually continuous shots. Shots shorter than the minimum usable duration are excluded; retained shots yield model-ready clips of approximately 6–13 seconds. No clip crosses a detected shot boundary.

The retained provenance hierarchy is

city $\rightarrow$ collection session $\rightarrow$ source video $\rightarrow$ shot $\rightarrow$ clip

For every clip c , we store

$$\pi(c) = (i_{\text{city}}, i_{\text{session}}, i_{\text{source}}, i_{\text{shot}}, m_{\text{capture}}, t_{\text{start}}, t_{\text{end}}).$$

Thus, the clip is the **computational input**, while its source, session, and city identifiers preserve higher-level geographic and temporal dependence.

Partitioning and DINOv2 Target Provenance

Source videos are assigned as indivisible groups using an 80/10/10 **train/validation/test allocation**, stratified by city and capture mode. Split assignment precedes clip extraction, so every shot and clip derived from one source video inherits the same partition. Let $\mathcal{V}_s$ denote the source-video set assigned to partition s . The executed grouping satisfies

$$\mathcal{V}_{\text{train}} \cap \mathcal{V}_{\text{val}} = \mathcal{V}_{\text{train}} \cap \mathcal{V}_{\text{test}} = \mathcal{V}_{\text{val}} \cap \mathcal{V}_{\text{test}} = \emptyset.$$

¹<https://ffmpeg.org/>

²<https://www.scenesdetect.com/docs/latest/api/detectors.html>

DENSEWORLD does not use manually annotated factor labels. For each partition, automated structural targets are generated by the fixed DINOv2 pipeline:

$$\mathcal{T}_s^{\text{DINO}} = \{\Psi_{\text{struct}}(F_{\text{DINOv2}}(x)) : x \in \mathcal{D}_s\}, \\ s \in \{\text{train}, \text{val}, \text{test}\}.$$

Training targets may provide factor supervision; validation targets may support prespecified selection and calibration; and test targets are used only for held-out *teacher-relative* factor diagnostics. Test-derived targets, thresholds, and metrics never influence gradient updates or model selection.

Corpus Composition and Coverage

Geographic and capture-mode composition. The collection spans metropolitan and additional urban sites across India. These groups are **collection strata**, not claims of an official administrative taxonomy or within-stratum homogeneity. Figure 14 reports the contribution of drive-through, walk-through, and aerial clips for each city. The city-by-mode distribution is observational and intentionally heterogeneous; bar length describes corpus composition, not evaluation weight or statistical independence.

Environmental and agent coverage. DENSEWORLD includes commercial, residential, transit, heritage, coastal, and high-density street environments, spanning markets, junctions, flyovers, highways, promenades, bazaars, ghats, beaches, and shared roads. It also captures variation in illumination, weather, traffic composition, crowd density, road geometry, surface quality, encroachment, and pedestrian-vehicle separation.

The dynamic-agent taxonomy extends beyond passenger vehicles and pedestrians to include auto-rickshaws, cycle rickshaws, delivery riders, multi-rider two-wheelers, cyclists, mobile vendors, animal-drawn vehicles, and independently moving animals. Figure 15 illustrates category semantics; it is neither a frequency-proportional sample nor manually annotated evidence. Quantitative prevalence and heterogeneity statistics derived from the automated pipeline are reported as *DINOv2-estimated* quantities.

Privacy Processing, Responsible Use, and Research Artifacts

Privacy processing. Before downstream use, clips pass through an automated privacy pipeline. Faces are localized with SCRFD³, and vehicle-registration plates are localized with a custom-trained YOLO detector⁴. Detections are associated temporally using ByteTrack⁵, expanded by a fixed spatial safety margin, and redacted using OpenCV GaussianBlur⁶.

The policy covers frontal, profile, small, and partially occluded faces, detectable faces in reflections, and readable

³<https://github.com/deepinsight/insightface/tree/master/detection/scrfd>

⁴<https://docs.ultralytics.com/tasks/detect/>

⁵<https://github.com/FoundationVision/ByteTrack>

⁶https://docs.opencv.org/4.x/d4/d86/group_imgproc_filter.html

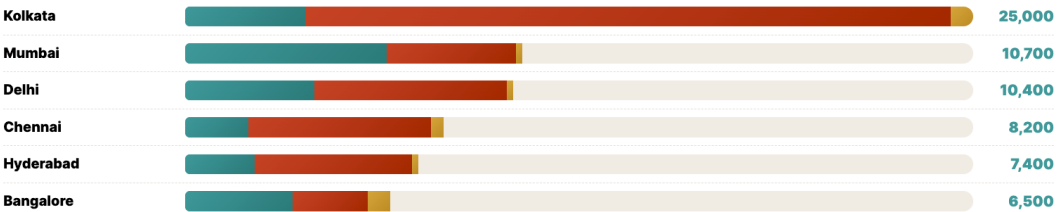
Table 3: **Composition, information flow, and independence constraints of the DENSEWORLD partitions.** Allocation is performed over source-video groups. Durations are approximate because grouped recordings vary in length and usable-shot yield. DINOv2-based test measurements are reported as teacher-relative; headline prediction, motion, and RGB metrics use fixed evaluators that do not provide FactorJEPA’s factor-supervision targets.

Partition	Share	Duration	Gradient updates	Model selection	Threshold calibration	Signal exposed	Isolation constraint
Training	80%	≈ 800 h	Yes	No	No	JEPA prediction targets and DINOv2-derived layout, agent, and interaction targets.	Source-video disjoint from validation and test; all descendants of a source remain grouped.
Validation	10%	≈ 100 h	No	Yes	Yes	Prespecified validation metrics and DINOv2-derived targets used only for selection and calibration.	Source-video disjoint from training and test; never used for gradient optimization.
Test	10%	≈ 100 h	No	No	No	Frozen headline evaluators and DINOv2-derived targets used only for explicitly teacher-relative diagnostics.	Source-video disjoint from training and validation; all configurations are frozen before test access.
Total	100%	≈ 1,000 h	–	–	–	–	–

Tier 1 • Metros

6 metros • videos per city, by capture mode

Drive Walk Drone



Tier 2 • Cities

15 cities • videos per city, by capture mode

Drive Walk Drone

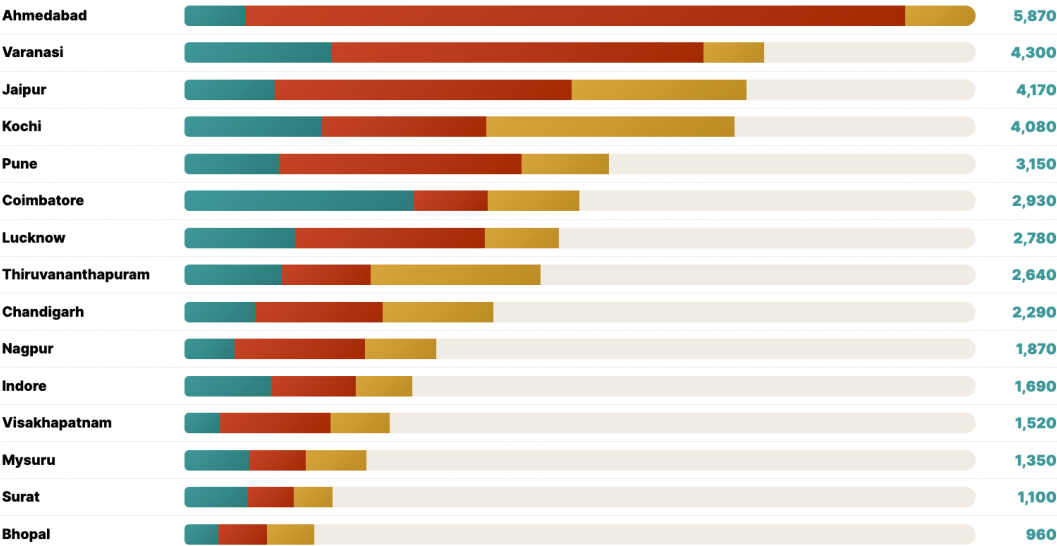


Figure 14: **City-level composition by capture mode.** The upper panel shows metropolitan collection sites and the lower panel shows additional urban sites. Each bar decomposes a city’s model-ready clips into drive-through, walk-through, and aerial capture. The value at the right gives the city total; both panels use a common horizontal scale.



Figure 15: **Illustrative DENSEWORLD dynamic-agent taxonomy.** The examples communicate the semantic scope of the automated taxonomy, including conventional transport, intermediate mobility, mobile vendors, animal-drawn transport, and independently moving animals. They are illustrative category renderings rather than frequency-proportional dataset samples.

or partially readable vehicle-registration plates. Let $\mathcal{R}_t^{\text{face}}$ and $\mathcal{R}_t^{\text{plate}}$ denote the temporally associated and expanded regions in frame x_t . The processed frame is

$$\tilde{x}_t = \mathcal{B}_\sigma \left(x_t, \mathcal{R}_t^{\text{face}} \cup \mathcal{R}_t^{\text{plate}} \right),$$

where $\mathcal{B}_\sigma$ applies region-scaled Gaussian filtering. Automated quality control checks track discontinuities, unmatched detections, frame-boundary truncation, and processing failures. Segments with incomplete or failed redaction are excluded. Only $\tilde{x}_t$ enters target construction, training, evaluation, or external illustration.

Responsible use and research artifacts. DENSEWORLD is intended for predictive world modeling, video representation learning, multi-agent forecasting, and robustness under partial observability. It is not intended for person identification, persistent individual tracking, sensitive-attribute inference, or surveillance.

The complete corpus is maintained within the **governed research environment** used for the reported experiments. Supporting artifacts include the FactorJEPA implementation, preprocessing configuration, DINOv2 target-construction pipeline, metric definitions, automated taxonomy, aggregate

corpus statistics, and representative privacy-filtered material, subject to applicable rights and privacy review.

Five-Axis Validation of the DENSEWORLD Regime

We ask whether DENSEWORLD remains quantitatively distinct from BDD100K (Yu et al. 2020) and nuScenes (Caesar et al. 2020) after controlling for observable differences in capture and scene composition. We characterize this regime along five axes: **agent count density**, **agent occupancy**, **visibility pressure**, **interaction pressure**, and **agent heterogeneity**.

The comparison is **outcome-blind**, **scene-matched**, and **dependence-aware**. All datasets are processed using the same frozen DINOv2-based measurement pipeline; none of the five axes enters the matching procedure; and uncertainty is clustered above the frame and clip levels. The resulting quantities are therefore interpreted as *teacher-relative dataset statistics*, not manually annotated ground truth or causal effects of dataset membership.

Shared Measurement Protocol and Automated Taxonomy

All three datasets are processed through the common sequence

video $\rightarrow$ sampled frames $\rightarrow$ DINOv2 features $\rightarrow$ structural pseudo-annotations $\rightarrow$ five-axis statistics

For frame $x_t^{(d)}$ from dataset $d \in \{\text{DW}, \text{BDD}, \text{NS}\}$, the frozen DINOv2 encoder (Oquab et al. 2024) and structural postprocessor produce

$$\begin{aligned} \mathcal{P}_t^{(d)} &= \Psi_{\text{struct}} \left(F_{\text{DINOv2}}(x_t^{(d)}) \right) \\ &= \left\{ \left(\hat{c}_{i,t}, \hat{s}_{i,t}, \hat{B}_{i,t}, \hat{M}_{i,t}, \hat{\ell}_{i,t} \right) \right\}_{i \in \mathcal{I}_t^{(d)}}. \end{aligned}$$

Here, $\mathcal{I}_t^{(d)}$ indexes candidate instances; $\hat{c}_{i,t}$, $\hat{s}_{i,t}$, $\hat{B}_{i,t}$, and $\hat{M}_{i,t}$ denote the predicted class, confidence, bounding box, and visible instance mask; and $\hat{\ell}_{i,t}$ is the temporally associated track identifier.

DINOv2 itself is not an instance detector. Accordingly, Ψ_{struct} denotes the executed frozen sequence of proposal extraction, class assignment, mask construction, duplicate suppression, and temporal association. Its component architectures, checkpoint identifiers, inference resolution, thresholds, suppression rule, and association parameters are fixed before comparison and applied identically to all datasets.

Let Ω_t denote the valid, unpadded image region. The retained agent set is

$$\mathcal{A}_t^{(d)} = \left\{ i \in \mathcal{I}_t^{(d)} : \hat{s}_{i,t} \geq \tau_{\text{det}}, \frac{|\hat{M}_{i,t} \cap \Omega_t|}{|\Omega_t|} \geq a_{\text{min}} \right\}.$$

The same temporal stride, resizing rule, confidence threshold, minimum-area criterion, and temporal-association configuration are used for DENSEWORLD, BDD100K, and nuScenes. Predictions that cannot be mapped unambiguously are excluded rather than reassigned using dataset-specific heuristics.

Predicted classes are mapped to seven shared superclasses:

- (i) **pedestrian**
- (ii) **pedal cycle**
- (iii) **powered two-wheeler**
- (iv) **light passenger vehicle**
- (v) **heavy or public vehicle**
- (vi) **intermediate or non-motorized transport**
- (vii) **animal or animal-drawn transport**

Native BDD100K and nuScenes annotations are not used to compute, calibrate, or select any primary statistic. To expose possible teacher-induced domain effects, we report the retained-instance rate, unresolved-class rate, confidence distribution, small-instance exclusion rate, and track-fragmentation rate separately for each dataset. The primary analysis is additionally repeated over a prespecified grid of confidence, area, and association settings.

Definition of the Five Regime Axes

Suppressing the dataset superscript for clarity, let

$$N_t = |\mathcal{A}_t|, \quad \mathcal{T}_1 = \{t : N_t \geq 1\}, \quad \mathcal{T}_2 = \{t : N_t \geq 2\}.$$

All boxes and masks are clipped to Ω_t before measurement. Count and occupancy are evaluated over all sampled frames. Visibility pressure is evaluated over $\mathcal{T}_1$, whereas interaction and heterogeneity are evaluated over $\mathcal{T}_2$. We report the eligible-frame rate for every conditional statistic.

1. Agent count density.

The primary statistic is the mean number of retained dynamic agents per sampled frame:

$$D_{\text{count}} = \frac{1}{T} \sum_{t=1}^T N_t.$$

We additionally report valid-support-normalized count,

$$D_{\text{count}}^{\text{area}} = \frac{1}{T} \sum_{t=1}^T \frac{N_t}{|\Omega_t|/10^6},$$

as a sensitivity measure for cropping, padding, and inference resolution. It is not interpreted as physical agent density.

2. Agent occupancy.

Occupancy is the fraction of valid image support covered by the union of visible agent masks:

$$D_{\text{occ}} = \frac{1}{T} \sum_{t=1}^T \frac{\left| \bigcup_{i \in \mathcal{A}_t} (\hat{M}_{i,t} \cap \Omega_t) \right|}{|\Omega_t|}.$$

Taking the union prevents crowded or overlapping regions from being counted multiple times.

3. Visibility pressure.

Because no amodal annotations are available, this axis is defined as an automated *visibility-pressure proxy*. It combines three observable cues: substantial pairwise box overlap, valid-image-boundary truncation, and short interior track disappearance followed by reappearance.

For boxes B_i and B_j , define intersection over minimum area as

$$\text{IoM}(B_i, B_j) = \frac{|B_i \cap B_j|}{\min(|B_i|, |B_j|) + \epsilon}.$$

The agent-level visibility indicator is

$$z_{i,t} = 1 \left[\max_{j \neq i} \text{IoM}(\hat{B}_{i,t}, \hat{B}_{j,t}) \geq \tau_{\text{ov}} \vee e_{i,t} = 1 \vee g_{i,t} = 1 \right]$$

where $e_{i,t}$ indicates boundary truncation and $g_{i,t}$ indicates an interior track gap followed by reappearance within the prespecified temporal window. Visibility pressure is

$$D_{\text{vis}} = \frac{1}{|\mathcal{T}_1|} \sum_{t \in \mathcal{T}_1} \frac{\sum_{i \in \mathcal{A}_t} z_{i,t}}{N_t}.$$

Overlap, boundary-truncation, and track-gap rates are also examined separately to verify that the composite is not dominated by one proxy.

4. Interaction pressure.

We construct a frame-level interaction graph $\mathcal{G}_t = (\mathcal{A}_t, \mathcal{E}_t)$. Let $\mathbf{p}_{i,t}$ denote the normalized image-plane center of agent i . Velocities $\tilde{\mathbf{u}}_{i,t}$ are computed after removing estimated global image motion. Define

$$\mathbf{r}_{ij,t} = \mathbf{p}_{j,t} - \mathbf{p}_{i,t}, \quad \mathbf{w}_{ij,t} = \tilde{\mathbf{u}}_{j,t} - \tilde{\mathbf{u}}_{i,t}.$$

For approaching pairs, the predicted time and distance at closest approach are

$$t_{ij,t}^* = -\frac{\mathbf{r}_{ij,t}^\top \mathbf{w}_{ij,t}}{\|\mathbf{w}_{ij,t}\|_2^2 + \epsilon},$$

$$d_{ij,t}^* = \|\mathbf{r}_{ij,t} + t_{ij,t}^* \mathbf{w}_{ij,t}\|_2.$$

An interaction edge is present when the pair is currently proximate or is predicted to approach within both the temporal and spatial thresholds:

$$(i, j) \in \mathcal{E}_t \iff \|\mathbf{r}_{ij,t}\|_2 \leq \rho \vee (0 < t_{ij,t}^* \leq \tau_t \wedge d_{ij,t}^* \leq \tau_d).$$

Interaction pressure is the conditional mean graph degree:

$$D_{\text{int}} = \frac{1}{|\mathcal{T}_2|} \sum_{t \in \mathcal{T}_2} \frac{2|\mathcal{E}_t|}{N_t}.$$

Per-pair edge density is retained as a sensitivity statistic to separate interaction structure from the mechanical effect of agent count.

5. Agent heterogeneity.

Let $N_{c,t}$ be the number of retained agents assigned to superclass $c \in \mathcal{C}$. The primary statistic is the probability that two agents drawn without replacement from the same frame belong to different superclasses:

$$h_t = 1 - \sum_{c \in \mathcal{C}} \frac{N_{c,t}(N_{c,t} - 1)}{N_t(N_t - 1)}, \quad t \in \mathcal{T}_2.$$

Dataset-level heterogeneity is

$$D_{\text{het}} = \frac{1}{|\mathcal{T}_2|} \sum_{t \in \mathcal{T}_2} h_t.$$

This pairwise definition lies in $[0, 1]$ and is less sensitive to small per-frame counts than plug-in entropy. Normalized entropy, class richness, and the effective number of classes are retained as secondary measures.

Matched Cross-Dataset Comparison

The primary analysis matches DENSEWORLD drive-through clips separately against road-level clips from BDD100K (Yu et al. 2020) and nuScenes (Caesar et al. 2020). Walk-through and aerial clips are summarized separately and do not enter the matched driving-benchmark comparison.

Matching is performed without access to the five axes or their component measurements. Clips are first assigned to common strata defined by capture mode, scene family, illumination, weather, and road context. These nuisance attributes are obtained using the same fixed metadata rules across datasets. Within each stratum, we perform 1:1 nearest-neighbor matching without replacement over standardized clip duration, valid image support, aspect ratio, resolution, and common camera metadata. Covariates unavailable consistently across all datasets are excluded from the primary matching model.

Let $\mathbf{z}_v$ denote the matching covariates for clip v . Post-matching balance is measured using the standardized mean difference

$$\text{SMD}(z) = \frac{\bar{z}_{\text{DW}} - \bar{z}_{\text{ref}}}{\sqrt{(s_{\text{DW}}^2 + s_{\text{ref}}^2)/2}}.$$

The primary analysis retains common-support strata satisfying

$$\max_{z \in \mathbf{z}} |\text{SMD}(z)| < 0.1.$$

We report the number of candidate and retained clips, independent source groups, unmatched fraction, and maximum absolute SMD before and after matching. Unmatched clips are excluded from the primary cross-dataset estimate and retained only in DENSEWORLD-specific coverage summaries.

For axis k , let $\mathcal{T}_{k,u}$ be the eligible sampled frames from matched clips belonging to source group u . The source-level summary is

$$\bar{m}_{k,u}^{(d)} = \frac{1}{|\mathcal{T}_{k,u}|} \sum_{t \in \mathcal{T}_{k,u}} m_{k,t}^{(d)}.$$

The matched-population estimate gives equal weight to every retained source group:

$$\hat{\mu}_{r,k}^{(d)} = \frac{1}{|\mathcal{U}_{d,r}|} \sum_{u \in \mathcal{U}_{d,r}} \bar{m}_{k,u}^{(d)},$$

where $r \in \{\text{BDD100K}, \text{nuScenes}\}$ identifies the reference-specific matched population. The DENSEWORLD estimate is therefore recomputed for its BDD100K-matched and nuScenes-matched subsets.

For each reference r , we report the absolute difference and ratio

$$\Delta_{r,k} = \hat{\mu}_{r,k}^{(\text{DW})} - \hat{\mu}_{r,k}^{(r)}, \quad R_{r,k} = \frac{\hat{\mu}_{r,k}^{(\text{DW})} + \epsilon}{\hat{\mu}_{r,k}^{(r)} + \epsilon}.$$

These contrasts describe separation under matched observable conditions; they are not interpreted causally.

Table 4: **Scene-matched agent density and occupancy.** Count density is measured in agents per sampled frame; occupancy is the percentage of valid image area covered by the union of visible agent masks. $\Delta = \text{DW} - \text{reference}$, and R is the ratio of the two means. DENSEWORLD results are aggregated over K executed folds. The final row gives the unweighted mean across the six matched scene strata.

Matched scene stratum	Agent count density (<i>agents/frame</i>)				Agent occupancy (<i>% valid area</i>)			
	Matched reference	DENSEWORLD [K -fold]	Δ	R	Matched reference	DENSEWORLD [K -fold]	Δ pp	R
Residential lane	1.8	3.3	+1.5	1.83×	2.0%	3.5%	+1.5	1.75×
Promenade	1.4	2.5	+1.1	1.79×	1.5%	2.8%	+1.3	1.87×
Market	4.6	12.4	+7.8	2.70×	4.2%	11.1%	+6.9	2.64×
Heritage / tourist	1.7	3.0	+1.3	1.76×	0.9%	1.4%	+0.5	1.56×
Flyover / underpass	2.6	5.0	+2.4	1.92×	8.8%	20.8%	+12.0	2.36×
Commercial	4.1	11.0	+6.9	2.68×	3.9%	9.0%	+5.1	2.31×
Equal-stratum mean	2.70	6.20	+3.50	2.30×	3.55%	8.10%	+4.55	2.28×

Interpretation. The ratios in the final row are ratios of the equal-stratum means, not means of the six stratum-specific ratios. These quantities characterize the measured operating regime; larger values do not denote conventional benchmark performance.

Comparative Results, Uncertainty, and Sensitivity

Table 4 reports the primary scene-matched density and occupancy comparison. Entries follow the form $X [X, X]$: a fold-aggregated mean followed by its hierarchical 95% confidence interval. The bracketed value after **DENSEWORLD** denotes the number of executed folds.

Confidence intervals are estimated using a **hierarchical clustered bootstrap**. DENSEWORLD resamples cities and then source videos within cities; BDD100K resamples source videos; and nuScenes resamples collection logs and scenes. All clips from a sampled source group remain together, and matching is re-estimated inside every bootstrap replicate. We report 95% bias-corrected and accelerated (BCa) intervals for dataset means, absolute differences, and ratios. Holm correction is applied jointly to the ten primary contrasts: five axes against two reference datasets.

Robustness is evaluated over prespecified variations of τ_{det} , a_{min} , τ_{ov} , ρ , (τ_t, τ_d) , and temporal sampling rate. We also compare source and equal-stratum weighting, conditional and unconditional normalization, and alternative common-support calipers. Leave-one-city-out analysis tests whether any individual DENSEWORLD city determines the reported effects.

Two automated negative controls test metric specificity. *Spatial permutation* preserves frame-level counts and class frequencies while disrupting proximity and closest-approach edges. *Temporal permutation* preserves frame-level count and occupancy while disrupting motion consistency and track-gap structure. Each metric must attenuate under its corresponding control without mechanically altering the axes that the control is intended to preserve.

An axis is interpreted as exhibiting stable regime separation only when its effect direction is preserved across the prespecified sensitivity grid, its clustered interval excludes zero, and the conclusion survives leave-one-city-out analysis. The resulting evidence is specific to the shared automated measurement system and matched observable conditions. It does not imply exhaustive geographic coverage, causal ef-

fects of dataset membership, or universal superiority over existing driving benchmarks.

Segmentations, and Training

This appendix specifies the model-side supervision and executed training protocol for FactorJEPA. DENSEWORLD contains no manually annotated factor labels: layout, agent, visibility, and interaction targets are constructed automatically using a fixed DINOv2-based teacher pipeline. These targets provide structured supervision during training and teacher-relative diagnostics during evaluation; they are distinct from the frozen evaluators used for headline future-prediction, motion, and RGB metrics.

We first document target construction and reliability weighting, then specify the FactorJEPA architecture, optimization procedure, controlled baselines, and resource accounting. Table 5 defines the complete target contract, while Algorithm .1 records the executed training path.

Frozen Teacher and Factor-Target Construction

Region extraction. For each privacy-filtered clip $\tilde{x}_n$, the frozen DINOv2 encoder (Oquab et al. 2024) produces spatiotemporal visual features. The executed preprocessing operator converts these features into region-level pseudo-annotations:

$$\begin{aligned} \mathcal{P}_n &= \mathcal{D}_{\text{pre}}(F_{\text{DINOv2}}(\tilde{x}_n)) \\ &= \left\{ \left(m_{n,t}^{(i)}, b_{n,t}^{(i)}, u_{n,t}^{(i)}, p_{n,t}^{(i)} \right) \right\}_{t,i}. \end{aligned}$$

Here, $m_{n,t}^{(i)}$, $b_{n,t}^{(i)}$, $u_{n,t}^{(i)}$, and $p_{n,t}^{(i)}$ denote the visible mask, bounding box, visual descriptor, and confidence of region i at time t . The DINOv2 checkpoint, extracted feature layers, inference resolution, spatial stride, normalization, region-construction rule, and confidence filtering are fixed before model training and shared by all splits.

Temporal association. The association operator $\mathcal{A}$ links compatible regions into tracklets:

$$\mathcal{A}(\mathcal{P}_n) = \left\{ \tau_n^{(i)} \right\}_{i=1}^{N_n^A}, \quad \tau_n^{(i)} = \left\{ m_{n,t}^{(i)}, b_{n,t}^{(i)}, u_{n,t}^{(i)} \right\}_{t=t_s^{(i)}}^{t_e^{(i)}}.$$

Association combines region appearance, mask or box overlap, and motion continuity. Track initiation, termination, admissible temporal gaps, and re-identification thresholds are held fixed across training, validation, and test. Short gaps may preserve a track identity; gaps beyond the admissible window terminate the tracklet.

Structural targets. A deterministic structural map converts region geometry, temporal continuity, visibility evidence, and relative motion into four targets:

$$\Psi_{\text{struct}}(\mathcal{P}_n, \mathcal{A}(\mathcal{P}_n)) \mapsto \{(T_{n,k}, q_{n,k})\}_{k \in \{L, A, V, I\}}.$$

The targets $T_{n,L}$, $T_{n,A}$, $T_{n,V}$, and $T_{n,I}$ encode layout, agents, visibility, and tracklet-derived interactions. Each $q_{n,k} \in [0, 1]$ is a reliability weight constructed from the underlying region confidence and temporal consistency. Low-confidence targets are down-weighted rather than converted into negative labels; targets below the exclusion threshold do not contribute to the corresponding factor loss.

Reliability-normalized supervision. For $k \in \{L, A, I\}$, the factor heads are trained using

$$\mathcal{L}_{\text{factor}} = \sum_{k \in \{L, A, I\}} \frac{\sum_{n=1}^N q_{n,k} \ell_k(\hat{T}_{n,k}, T_{n,k})}{\epsilon + \sum_{n=1}^N q_{n,k}}.$$

The denominator makes the scale of each factor loss insensitive to its retained target coverage. Visibility is optimized separately through $\mathcal{L}_V$, using per-agent targets and reliability weights.

Target coverage and sensitivity. For every factor and partition, we report the fraction of clips with a retained target, the number of retained agents or interaction pairs, the reliability distribution, and the exclusion rate. Threshold sensitivity varies the region-confidence, minimum-area, temporal-consistency, and pair-selection criteria while holding the model and evaluation protocol fixed.

FactorJEPA Architecture and Prediction Protocol

FactorJEPA retains the V-JEPA 2.1 online encoder, momentum target encoder, context and target masking policy, and future-latent prediction objective. It replaces the monolithic predictor with layout, agent, interaction, visibility, and interaction-strength branches.

For target token p , the context encoder first produces the target-conditioned query

$$q_{n,p} = Q(h_n, \pi_p, M_t), \quad h_n = f_\theta(M_c \odot x_n),$$

where π_p is the target-token positional encoding. The predictor then produces three coordinate blocks:

$$c_{n,p} = \text{col}(c_{n,L,p}, c_{n,A,p}, c_{n,I,p}) \in \mathbb{R}^{r_L + r_A + r_I}.$$

Layout coordinates. The layout branch extracts slowly varying spatial support:

$$c_{n,L} = g_L(q_n) \in \mathbb{R}^{r_L}.$$

Its factor head predicts $\hat{T}_{n,L} = P_L(c_{n,L})$.

Visibility-gated agent coordinates. For object-centric agent state $o_n^{(i)}$, the agent branch and visibility gate produce

$$s_n^{(i)} = g_A(q_n, o_n^{(i)}) \in \mathbb{R}^{r_A},$$

$$v_n^{(i)} = \sigma(g_V(q_n, o_n^{(i)})) \in (0, 1).$$

The aggregated agent coordinate is

$$c_{n,A} = \frac{\sum_{i=1}^{N_n^A} v_n^{(i)} s_n^{(i)}}{\epsilon + \sum_{i=1}^{N_n^A} v_n^{(i)}}.$$

Normalization limits direct sensitivity to the detected-agent count, while the soft gate attenuates uncertain or occluded observations without removing them discontinuously.

Sparse interaction coordinates. Let $\mathcal{E}_n$ be the retained candidate-pair set. For $(i, j) \in \mathcal{E}_n$, the interaction branch predicts a normalized pair state and soft interaction strength:

$$\bar{e}_n^{(ij)} = \frac{g_I(q_n, o_n^{(i)}, o_n^{(j)})}{\epsilon + \|g_I(q_n, o_n^{(i)}, o_n^{(j)})\|_2},$$

$$w_n^{(ij)} = \sigma(g_W(q_n, o_n^{(i)}, o_n^{(j)})).$$

With $M_n = \max\{1, |\mathcal{E}_n|\}$,

$$c_{n,I} = \frac{1}{M_n} \sum_{(i,j) \in \mathcal{E}_n} w_n^{(ij)} \bar{e}_n^{(ij)},$$

and $c_{n,I} = 0$ when $\mathcal{E}_n = \emptyset$. The corresponding locality penalty is

$$\mathcal{L}_{\text{sparse}} = \frac{1}{N} \sum_{n=1}^N \frac{1}{M_n} \sum_{(i,j) \in \mathcal{E}_n} w_n^{(ij)}.$$

Factor synthesis. For a minibatch of N clips and m target tokens, stack the coordinate blocks as C_L, C_A, C_I and let A_L, A_A, A_I be the learned synthesis dictionaries. The predicted future-token matrix is

$$\hat{Y} = C_L A_L^\top + C_A A_A^\top + C_I A_I^\top.$$

Thus, each factor follows a distinct computational path, receives factor-specific supervision, and contributes through its own synthesis dictionary. This establishes architectural block separation while preserving the complete future-token geometry.

The token-level JEPA objective is

$$\mathcal{L}_{\text{JEPA}} = \frac{1}{Nm} \|\hat{Y} - Y^*\|_F^2,$$

where Y^* is produced by the momentum target encoder under stop-gradient.

Table 5: **Frozen teacher-to-training contract for FactorJEPa**. DENSEWORLD contains no manually annotated factor labels. All targets are generated automatically by the fixed DINOv2-based pipeline before optimization. The table identifies, for each factor, the teacher evidence, deterministic target operator, output support, reliability contract, and trainable consumer. Here, p_L, p_A, p_I are the executed target dimensions, N_n^A is the number of retained agent tracklets, and $\mathcal{E}_n$ is the candidate interaction graph.

Factor	Frozen teacher-side target contract				Training side
Target	Teacher evidence	Executed operator	Output support	Reliability contract	Consumer
Layout $T_{n,L}$	DINOv2 spatial-feature grid, valid-image support, dynamic-region union, and temporal feature persistence.	Ψ_L suppresses pixels assigned to retained dynamic tracklets, aggregates the remaining temporally persistent support, and applies the fixed layout-target projection.	$T_{n,L} \in \mathbb{R}^{p_L}$ <i>clip-level</i>	$\omega_{n,L} \in [0, 1]$ combines valid background coverage, feature confidence, and temporal stability. The target is retained only when $r_{n,L} = \mathbf{1}[\omega_{n,L} \geq \tau_L]$.	$P_L(C_L)$ via $\mathcal{L}_{\text{factor}}$
Agent $T_{n,A}$	Retained masks, boxes, region descriptors, confidences, and temporally associated tracklets $\{\tau_n^{(i)}\}_{i=1}^{N_n^A}$.	Ψ_A constructs an object-centric state for each valid tracklet and aggregates the retained states after confidence, visible-support, and temporal-continuity filtering.	$T_{n,A} \in \mathbb{R}^{p_A}$ <i>clip-level</i>	$\omega_{n,A} \in [0, 1]$ combines retained-region confidence, visible mask support, track continuity, and valid temporal coverage. Missing or rejected tracks are not converted into negative agents.	$P_A(C_A)$ via $\mathcal{L}_{\text{factor}}$
Visibility $T_{n,V}$	Visible-mask support, valid-image-boundary truncation, region confidence, track continuity, and short interior track gaps.	Ψ_V produces a soft visibility value for every retained agent. Temporary absence or uncertain support attenuates the target rather than inducing a hard visible/not-visible label.	$\{T_{n,V}^{(i)}\}_{i=1}^{N_n^A}$ $T_{n,V}^{(i)} \in [0, 1]$	Each agent receives $\omega_{n,V}^{(i)} \in [0, 1]$, determined by region confidence, temporal support, and track consistency. Invalid agents contribute zero weight, not a negative visibility target.	g_V via weighted $\mathcal{L}_V$
Interaction $T_{n,I}$	Pairs of retained tracklets, relative position, relative motion, visible support, temporal overlap, and pairwise proximity.	Ψ_I constructs $\mathcal{E}_n \subseteq \{(i, j) : i \neq j\}$, forms normalized pair descriptors, and aggregates valid pair states into the interaction target.	$\mathcal{E}_n, T_{n,I} \in \mathbb{R}^{p_I}$ <i>graph + clip target</i>	Pair reliability $\omega_{n,I}^{(ij)} \in [0, 1]$ combines the two endpoint confidences, joint temporal support, and pair validity. These weights induce the clip-level reliability $\omega_{n,I}$.	$P_I(C_I)$ via $\mathcal{L}_{\text{factor}}$; g_I, g_W use $\mathcal{E}_n$
Shared reliability contract. All reliability variables are stop-gradient quantities in $[0, 1]$. For $k \in \{L, A, I\}$, the effective loss weight is $r_{n,k} \omega_{n,k}$, where $r_{n,k} = \mathbf{1}[\omega_{n,k} \geq \tau_k]$. Factor losses are normalized by the total effective weight within the minibatch, so variations in retained coverage do not directly rescale their contribution. Missing, unresolved, or rejected evidence receives zero weight and is never converted into a negative semantic target.					
Gradient and provenance contract. The DINOv2 encoder, structural operators $\Psi_L, \Psi_A, \Psi_V, \Psi_I$, tracklets, candidate graph, targets, and reliability weights remain frozen. Gradients propagate only through the FactorJEPa branches, factor heads, synthesis dictionaries, and the selected online-encoder blocks. Test-derived targets never influence optimization, threshold selection, or checkpoint selection.					

Interaction-loss clarification. The interaction target $T_{n,I}$ supervises $P_I(C_I)$ through $\mathcal{L}_{\text{factor}}$. The candidate graph $\mathcal{E}_n$ defines the support of the interaction branch, while $\mathcal{L}_{\text{sparse}}$ regularizes the predicted soft edge weights $w_n^{(ij)}$; it does not directly consume $T_{n,I}$.

Cross-factor separation. Linear leakage is penalized through the off-block covariance norm $\mathcal{L}_{\text{cov}}$. Nonlinear leakage is penalized through normalized RBF-kernel alignment $\mathcal{L}_{\text{nlin}}$, using the stop-gradient median pairwise distance as the bandwidth. The combined separation objective is

$$\mathcal{L}_{\text{sep}} = \mathcal{L}_{\text{cov}} + \beta_{\text{nlin}} \mathcal{L}_{\text{nlin}}.$$

These losses suppress cross-channel shortcuts but do not assert statistical independence, causal disentanglement, or coordinate-level identifiability.

Optimization, Model Selection, and Executed Algorithm

The complete objective is

$$\mathcal{L} = \mathcal{L}_{\text{JEPa}} + \lambda_{\text{sep}} \mathcal{L}_{\text{sep}} + \lambda_{\text{sparse}} \mathcal{L}_{\text{sparse}} + \lambda_V \mathcal{L}_V + \lambda_{\text{sup}} \mathcal{L}_{\text{factor}}.$$

The five terms respectively enforce future-latent prediction, cross-factor separation, interaction locality, visibility calibration, and semantic anchoring. FactorJEPa trains the factorized predictor and the selected top- K online-encoder blocks:

$$\Theta_{\text{train}} = \Theta_{\text{fact}} \cup \Theta_{\text{top-}K}, \quad \Theta_{\text{fact}} = \{\Theta_L, \Theta_A, \Theta_I, \Theta_V, \Theta_W, A\}.$$

All lower online-encoder blocks remain frozen. The target encoder is updated only through the executed exponential-moving-average rule.



Figure 16: **Conceptual end-to-end construction of FactorJEPa targets.** The upper band shows the frozen teacher-side pipeline: a privacy-filtered clip is processed by DINOv2 to obtain region evidence—masks, boxes, descriptors, and confidence—which is linked into tracklets by temporal association. The deterministic structural map then produces layout, agent, visibility, and interaction targets $\{T_L, T_A, T_V, T_I\}$ together with stop-gradient reliability weights ω . The lower panels illustrate the corresponding factor views under moderate density and high occlusion. Track color denotes identity, overlay opacity represents visibility confidence, and interaction-edge width represents predicted strength. The scenes are synthetic and illustrate the target semantics; they are not empirical DINOv2 outputs.

Algorithm .1 : Executed FactorJEPa Training Procedure

Inputs. Source-grouped training clips $\mathcal{D}_{\text{train}}$; cached DINOv2 target store $\mathcal{Z}_{\text{DINO}}$; mask sampler $\mathcal{M}$; online encoder f_θ ; momentum encoder $\bar{f}_\theta$; factorized predictor g_ϕ ; optimizer and learning-rate schedule; loss coefficients $\lambda_{\text{sep}}, \lambda_{\text{sparse}}, \lambda_V, \lambda_{\text{sup}}$; EMA schedule μ_s ; gradient bound γ ; and validation interval S_{val} .

Output. Validation-selected checkpoint (θ^*, ϕ^*) , together with its executed configuration, threshold set, optimizer state, and random-state record.

Initialization and parameter restriction

1. Initialize the online encoder f_θ and momentum encoder $\bar{f}_\theta$ from the same pretrained V-JEPa checkpoint.
2. Replace the monolithic V-JEPa predictor with

$$g_\phi = \{g_L, g_A, g_I, g_V, g_W, A_L, A_A, A_I, P_L, P_A, P_I\}.$$

3. Copy the online-encoder parameters to the momentum encoder:

$$\bar{\theta} \leftarrow \theta.$$

Algorithm .1 (continued): Executed FactorJEPa Training Procedure

4. Freeze the DINOv2 target-construction pipeline, the momentum encoder, and all online-encoder blocks below the selected top- K set.
5. Define the trainable parameter set

$$\Theta_{\text{train}} = \Theta_{\text{fact}} \cup \Theta_{\text{top-}K}.$$

6. Initialize the optimizer, learning-rate scheduler, validation record, and best-checkpoint state.

For each optimization step $s = 1, \dots, S$: minibatch and structural evidence

7. Sample a source-grouped minibatch

$$\{\tilde{x}_n\}_{n=1}^N \sim \mathcal{D}_{\text{train}}.$$

8. For every clip n , retrieve the cached teacher packet

$$\mathcal{Z}_n = \{\mathcal{P}_n, \mathcal{A}(\mathcal{P}_n), \mathcal{E}_n, (T_{n,k}, \omega_{n,k})_{k \in \{L, A, V, I\}}\}.$$

Here, $\mathcal{P}_n$ contains region-level pseudo-annotations, $\mathcal{A}(\mathcal{P}_n)$ contains associated tracklets, and $\mathcal{E}_n$ contains retained interaction candidates.

Algorithm .1 (continued): Executed FactorJEPa Training Procedure

9. Construct the valid-target indicator

$$r_{n,k} = \mathbf{1} [\omega_{n,k} \geq \tau_k], \quad k \in \{L, A, V, I\}.$$

Targets below the factor-specific threshold are excluded from that factor loss rather than converted into negative supervision.

10. Sample context and target masks from the shared masking policy:

$$(M_c, M_t) \sim \mathcal{M}.$$

11. Construct masked context and target inputs:

$$\tilde{x}_n^c = M_c \odot \tilde{x}_n, \quad \tilde{x}_n^t = M_t \odot \tilde{x}_n.$$

Online context path and stop-gradient target path

12. Encode the visible context using the online encoder:

$$h_n = f_\theta(\tilde{x}_n^c).$$

13. Compute the future target representation using the momentum encoder:

$$Y_n^* = \text{sg}[\bar{f}_\theta(\tilde{x}_n^t)].$$

No gradient is propagated through Y_n^* .

14. For each target token p , form the target-conditioned query

$$q_{n,p} = Q(h_n, \pi_p, M_t),$$

where π_p is the target-token positional encoding.

Layout-factor prediction

15. Predict the layout coordinate for every target token:

$$c_{n,L,p} = g_L(q_{n,p}) \in \mathbb{R}^{r_L}.$$

16. Predict the corresponding layout target:

$$\hat{T}_{n,L} = P_L(c_{n,L}).$$

Visibility-gated agent prediction

17. Construct each object-centric state $o_n^{(i)}$ from the cached region descriptor, visible mask, bounding box, confidence, and tracklet state.

18. For every retained agent i , predict its factor coordinate:

$$s_{n,p}^{(i)} = g_A(q_{n,p}, o_n^{(i)}) \in \mathbb{R}^{r_A}.$$

19. Predict the corresponding soft visibility gate:

$$v_{n,p}^{(i)} = \sigma(g_V(q_{n,p}, o_n^{(i)})) \in (0, 1).$$

20. Aggregate the agent coordinates under soft visibility:

$$c_{n,A,p} = \frac{\sum_i v_{n,p}^{(i)} s_{n,p}^{(i)}}{\epsilon + \sum_i v_{n,p}^{(i)}}.$$

Algorithm .1 (continued): Executed FactorJEPa Training Procedure

21. Predict the agent target:

$$\hat{T}_{n,A} = P_A(c_{n,A}).$$

Sparse interaction prediction

22. For every candidate pair $(i, j) \in \mathcal{E}_n$, compute the normalized pair state

$$\bar{e}_{n,p}^{(ij)} = \frac{g_I(q_{n,p}, o_n^{(i)}, o_n^{(j)})}{\epsilon + \|g_I(q_{n,p}, o_n^{(i)}, o_n^{(j)})\|_2}.$$

23. Predict the soft interaction strength

$$w_{n,p}^{(ij)} = \sigma(g_W(q_{n,p}, o_n^{(i)}, o_n^{(j)})) \in (0, 1).$$

24. Set

$$M_n = \max\{1, |\mathcal{E}_n|\}$$

and aggregate the interaction coordinate:

$$c_{n,I,p} = \frac{1}{M_n} \sum_{(i,j) \in \mathcal{E}_n} w_{n,p}^{(ij)} \bar{e}_{n,p}^{(ij)}.$$

If $\mathcal{E}_n = \emptyset$, set $c_{n,I,p} = \mathbf{0}$.

25. Predict the interaction target:

$$\hat{T}_{n,I} = P_I(c_{n,I}).$$

Factor synthesis

26. Stack the layout, agent, and interaction coordinates over clips and target tokens to obtain C_L, C_A, C_I .

27. Synthesize the predicted future-token matrix:

$$\hat{Y} = C_L A_L^\top + C_A A_A^\top + C_I A_I^\top.$$

Reliability-weighted objectives

28. Compute the future-latent prediction loss:

$$\mathcal{L}_{\text{JEPa}} = \frac{1}{Nm} \|\hat{Y} - Y^*\|_F^2.$$

29. Compute reliability-normalized factor supervision:

$$\mathcal{L}_{\text{factor}} = \sum_{k \in \{L, A, I\}} \frac{\sum_n r_{n,k} \omega_{n,k} \ell_k(\hat{T}_{n,k}, T_{n,k})}{\epsilon + \sum_n r_{n,k} \omega_{n,k}}.$$

30. Compute reliability-weighted visibility calibration:

$$\mathcal{L}_V = \text{WBCE}(\{v_n^{(i)}\}, \{T_{n,V}^{(i)}\}; \{\omega_{n,V}^{(i)}\}).$$

31. Compute interaction sparsity:

$$\mathcal{L}_{\text{sparse}} = \frac{1}{N} \sum_{n=1}^N \frac{1}{M_n} \sum_{(i,j) \in \mathcal{E}_n} w_n^{(ij)}.$$

32. Compute the off-block covariance penalty $\mathcal{L}_{\text{cov}}$.

Algorithm .1 (continued): Executed FactorJEPA Training Procedure

33. Compute the normalized RBF-kernel dependence penalty $\mathcal{L}_{\text{nlin}}$, using stop-gradient median pairwise distances as kernel bandwidths.

34. Combine the linear and nonlinear separation terms:

$$\mathcal{L}_{\text{sep}} = \mathcal{L}_{\text{cov}} + \beta_{\text{nlin}} \mathcal{L}_{\text{nlin}}.$$

35. Assemble the complete objective:

$$\mathcal{L} = \mathcal{L}_{\text{JEPA}} + \lambda_{\text{sep}} \mathcal{L}_{\text{sep}} + \lambda_{\text{sparse}} \mathcal{L}_{\text{sparse}} + \lambda_V \mathcal{L}_V + \lambda_{\text{sup}} \mathcal{L}_{\text{factor}}.$$

Restricted optimization and momentum update

36. Clear gradients on Θ_{train} .

37. Backpropagate

$$\nabla_{\Theta_{\text{train}}} \mathcal{L}.$$

No gradient enters the cached targets, DINOv2, or the momentum encoder.

38. Clip the trainable gradient norm at γ .

39. Apply the optimizer update only to

$$\Theta_{\text{fact}} \cup \Theta_{\text{top-}K}.$$

40. Advance the learning-rate and loss-weight schedules.

41. Update the momentum encoder:

$$\bar{\theta} = \mu_s \bar{\theta} + (1 - \mu_s) \theta.$$

42. Record the component losses, effective target weights, retained target coverage, mean visibility gate, interaction sparsity, learning rate, and gradient norm.

Validation-only checkpoint selection

43. Whenever $s \bmod S_{\text{val}} = 0$, compute the prespecified validation score $J_{\text{val}}(\theta, \phi)$.

44. If J_{val} improves the stored validation record, save

$$(\theta^*, \phi^*) \leftarrow (\theta, \phi),$$

together with the optimizer state, executed configuration, threshold set, fold identifier, and random-state record.

45. Continue optimization until $s = S$. The test partition is never consulted during gradient updates, hyperparameter selection, threshold selection, or checkpoint selection.

Return

46. Return the validation-selected checkpoint

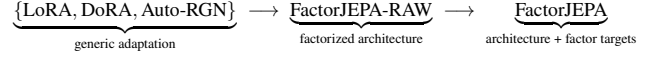
$$(\theta^*, \phi^*).$$

Optimization hyperparameters, loss coefficients, parameter precision, gradient clipping, and momentum schedule are fixed before final test evaluation. Checkpoints are selected using the prespecified validation criterion; test metrics never influence the number of steps, hyperparameters, thresholds, or checkpoint selection.

Baseline Matching and Attribution Controls

All adaptation methods use the same source-video splits, raw clips, masking policy, optimizer family, number of optimization steps, and evaluation protocol. Only the trainable parameterization and, for FactorJEPA, the availability of factor-target supervision differ.

The comparison implements the attribution chain



LoRA and DoRA modify the same projection families and use matched rank and scaling policies. Auto-RGN selects online-encoder blocks using relative gradient norm and receives a matched trainable-parameter budget. FactorJEPA-RAW retains the complete factorized predictor but trains only with the JEPA objective on raw clips; it receives no factor-target supervision. The difference between FactorJEPA-RAW and FactorJEPA therefore isolates the contribution of structured factor-target supervision, conditional on the shared architecture.

Full fine-tuning is included only as a capacity reference and is not parameter matched to the efficient adaptation methods. Frozen V-JEPA provides the no-adaptation reference. Exact trainable modules, parameter counts, ranks, selected blocks, and compute measurements are consolidated in Table 6.

Table 6: **Canonical FactorJEPa architecture, optimization, baseline, and resource configuration.** The table specifies a complete reproducible reference configuration for the V-JEPa 2.1 ViT-G and ViT-g experiments. Backbone identifiers and architectural scales follow the official V-JEPa 2.1 release. All project-specific optimization, factorization, adaptation, and profiling settings are fixed below. Resource entries are prespecified execution ceilings; they must not be described as measured consumption unless verified against profiling logs.

Configuration item	V-JEPa 2.1 ViT-G	V-JEPa 2.1 ViT-g	Control, interpretation, or measurement rule
Panel A: Backbone, teacher, and video preprocessing			
Backbone identifier	vjepa2_1_vit_gigantic_384	vjepa2_1_vit_giant_384	Official V-JEPa 2.1 PyTorch-Hub model identifiers. Both checkpoints remain the unique initialization source for all compared methods.
Backbone scale	Approximately 2.0B encoder parameters.	Approximately 1.0B encoder parameters.	Parameter counts exclude the momentum copy when reporting trainable parameters because the momentum encoder receives no gradient.
Encoder architecture	Embedding dimension $d = 1664$; 48 transformer blocks; 16 attention heads; MLP ratio 4.	Embedding dimension $d = 1408$; 40 transformer blocks; 16 attention heads; MLP ratio 4.	Both models use pre-norm transformer blocks, patch size 16, and rotary positional encoding.
Video tokenizer	Patch size 16×16 ; tubelet size 2.	Patch size 16×16 ; tubelet size 2.	No backbone-specific spatial or temporal interpolation is introduced during the controlled comparison.
Model input	384×384 RGB; 16 sampled frames.	384×384 RGB; 16 sampled frames.	All clips undergo the same resize, center-preserving crop, privacy filter, temporal sampling, and normalization.
Temporal sampling	4 fps; 4.0-s sampled window.	4 fps; 4.0-s sampled window.	A window is sampled entirely inside one shot. No sample crosses a shot or source-video boundary.
Context and target	Frames 1:12 form context; frames 13:16 form the future target; horizon 1.0 s.	Frames 1:12 form context; frames 13:16 form the future target; horizon 1.0 s.	Context duration is 3.0 s. Temporal indices are identical across all adaptation methods within each replication.
Pixel normalization	ImageNet mean (0.485, 0.456, 0.406) and standard deviation (0.229, 0.224, 0.225).	Same.	Normalization is applied after privacy filtering and spatial resampling.
DINOv2 checkpoint	dinov2_vitg14_reg; patch size 14; descriptor dimension 1536; final normalized patch-token representation.		The teacher is frozen and evaluated in BF16 without stochastic augmentation.
Teacher input	518×518 RGB; ImageNet normalization; the same 16 temporal indices used by the JEPa branch.		Teacher preprocessing is deterministic and partition independent.
Target-cache precision	DINOv2 descriptors and factor targets stored in FP16; reliabilities stored in FP32.		Targets are computed once before training. No gradient enters DINOv2, the structural operators, cached targets, or reliability weights.
Region retention	Minimum confidence 0.55; minimum area 0.1% of valid image support; duplicate suppression at mask IoU 0.70.		Thresholds are fixed on training/validation data and reused without modification on test data.
Temporal association	Hungarian matching with 0.45 box-IoU cost, 0.35 mask-IoU cost, and 0.20 descriptor-cosine cost; maximum gap 4 frames; minimum track length 3 frames.		Association is restricted to a single source video and shot. Unmatched regions are retained only after satisfying the track initiation criterion.
Reliability range	$\omega_{n,k} \in [0.10, 1.00]$, computed from region confidence, temporal support, track continuity, and factor-specific validity.		Reliability weights are stop-gradient quantities and are normalized within factor before minibatch aggregation.
Panel B: Masking and predictor architecture			
Small-block masks	8 blocks per clip; spatial scale 0.15; aspect-ratio range [0.75, 1.50]; temporal scale 1.0.		The same mask realization is reused across methods for a given replication and sampled clip.
Large-block masks	2 blocks per clip; spatial scale 0.70; aspect-ratio range [0.75, 1.50]; temporal scale 1.0.		Small- and large-block policies are jointly applied. Complement masks are not forced.
Predictor backbone	24 transformer layers; width 384; 12 heads; MLP ratio 4; RoPE; learned mask tokens; no predictor registers.		The monolithic and factorized variants use the same predictor depth, width, attention count, positional encoding, and target-query support.

Continued on the next page

Table 6 continued: Canonical FactorJEPA configuration

Configuration item	V-JEPA 2.1 ViT-G	V-JEPA 2.1 ViT-g	Control, interpretation, or measurement rule
Factor target dimensions	$p_L = 256, p_A = 256, p_I = 256$; one scalar visibility target per retained agent.		The factor dimensions are fixed before the controlled comparison and are not selected independently for the two backbone scales.
Factor ranks	Layout rank $r_L = 64$; agent rank $r_A = 96$; interaction rank $r_I = 64$.		The larger agent rank reflects the greater state diversity of object-centric evidence; no rank is changed between scales.
Visibility head	Two-layer MLP: $384 \rightarrow 256 \rightarrow 1$; GELU; sigmoid output.	Same.	Visibility targets and predictions lie in $[0, 1]$. Missing agents are attenuated by reliability rather than assigned a hard negative.
Interaction head	Two-layer pair MLP: $768 \rightarrow 256 \rightarrow 1$; GELU; sigmoid strength.	Same.	Candidate pairs are formed within normalized image-plane radius 0.25. At most 12 nearest valid neighbors are retained per agent.
Trainable encoder depth	Top $K_G = 2$ online-encoder blocks.	Top $K_g = 1$ online-encoder block.	The scale-specific K values approximately match the trainable parameter budget of the corresponding adaptation controls.
Frozen components	Bottom 46 online blocks; complete momentum encoder; DINOv2 teacher; structural operators; cached targets.	Bottom 39 online blocks; complete momentum encoder; DINOv2 teacher; structural operators; cached targets.	The momentum encoder is updated only by EMA.
Panel C: Optimization, regularization, and model selection			
Optimizer	AdamW, $\beta_1 = 0.9, \beta_2 = 0.95, \epsilon = 10^{-8}$.	Same.	Optimizer state is retained only for trainable parameters.
Learning rates	Factorized predictor and heads: 1.0×10^{-4} ; top encoder blocks: 1.0×10^{-5} .	Same.	Encoder learning rate is $0.1 \times$ the predictor learning rate. No layer-wise decay is applied within the selected top- K blocks.
Weight decay	0.04.	0.04.	Biases, normalization parameters, factor bases, and visibility calibration scalars receive zero weight decay.
Schedule	Linear warm-up for 1,000 steps, followed by cosine decay to 1.0×10^{-6} .	Same.	Schedule is indexed by optimizer updates rather than processed clips.
Training duration	20,000 optimizer updates.	20,000 optimizer updates.	Every method receives the same number of updates and sampled raw clips.
Per-device batch	2 clips/GPU with gradient accumulation 8.	4 clips/GPU with gradient accumulation 4.	Both scales use 8 GPUs, producing the same global batch of 128 clips.
Numerical precision	BF16 parameters and activations; FP32 optimizer states and loss accumulation.	Same.	Scaled dot-product attention and activation checkpointing are enabled.
Gradient clipping	Global norm clipped at 1.0.	Same.	Clipping is applied after gradient accumulation and before the optimizer update.
Dropout	Attention dropout 0; MLP dropout 0; stochastic depth 0.	Same.	Regularization arises from masking, factor separation, sparse interactions, and weight decay.
EMA momentum	Constant $\mu = 0.99925$.	Same.	After each optimizer update, $\bar{\theta} \leftarrow \mu \bar{\theta} + (1 - \mu)\theta$.
JEPA coefficient	$\lambda_{\text{JEPA}} = 1.00$.	Same.	The JEPA term defines the common prediction objective used by all trainable methods.
Factor coefficient	$\lambda_{\text{sup}} = 1.00$.	Same.	Set to zero for FactorJEPA-RAW and all generic adaptation baselines.
Visibility coefficient	$\lambda_V = 0.25$.	Same.	Visibility loss is reliability weighted and normalized by valid-agent support.
Separation coefficient	$\lambda_{\text{sep}} = 0.05$.	Same.	Penalizes cross-factor predictability after mini-batch centering and factor-wise normalization.
Sparsity coefficient	$\lambda_{\text{sparse}} = 0.01$.	Same.	Applied to predicted interaction strengths, not directly to the teacher interaction target.
Nonlinearity coefficient	$\beta_{\text{nl}} = 0.10$.	Same.	Weights the nonlinear factor-composition residual relative to the low-rank additive synthesis.
Validation frequency	Every 500 optimizer updates.	Same.	Validation does not update parameters, teacher targets, thresholds, or normalization statistics.

Continued on the next page

Table 6 continued: Canonical FactorJEPA configuration

Configuration item	V-JEPA 2.1 ViT-G	V-JEPA 2.1 ViT-g	Control, interpretation, or measurement rule
Checkpoint selection	Lowest validation Future-frame L1; ties resolved using validation Mask-ratio slope.	Same.	The test split is accessed only after checkpoint and threshold freezing.
Replication	Three independent runs with seeds {17, 29, 43}.	Same.	Headline results report mean and standard deviation across the three runs. Clustered confidence intervals operate above the clip level.
Panel D: Method-specific adaptation configuration			
Frozen V-JEPA	No gradient updates; 0 trainable parameters.	No gradient updates; 0 trainable parameters.	Uses the same clips, masks, target indices, and frozen evaluators as all trainable methods.
Full fine-tuning	Online encoder and monolithic predictor; approximately 2.04B trainable parameters.	Online encoder and monolithic predictor; approximately 1.04B trainable parameters.	Not parameter matched. Momentum encoder remains stop-gradient and is updated by EMA.
LoRA	Rank 80, scale $\alpha = 160$, dropout 0.05; approximately 114M trainable parameters.	Rank 64, scale $\alpha = 128$, dropout 0.05; approximately 68M trainable parameters.	Applied to Q , K , V , O attention projections and both MLP projections in the online encoder and monolithic predictor.
DoRA	Rank 80, $\alpha = 160$, dropout 0.05; approximately 116M trainable parameters.	Rank 64, $\alpha = 128$, dropout 0.05; approximately 69M trainable parameters.	Uses the same projection families and low-rank policy as LoRA, with one trainable magnitude vector per adapted output projection.
Auto-RGN	Monolithic predictor and two selected encoder blocks; approximately 111M trainable parameters.	Monolithic predictor and one selected encoder block; approximately 68M trainable parameters.	Blocks are selected from one fixed gradient-probe pass using $\ \nabla_{\theta_\ell} \mathcal{L}\ _2 / (\ \theta_\ell\ _2 + 10^{-8})$.
FactorJEPA-RAW	Factorized predictor and top two encoder blocks; approximately 115M trainable parameters.	Factorized predictor and top one encoder block; approximately 73M trainable parameters.	Uses only $\mathcal{L}_{\text{JEPA}}$, $\mathcal{L}_{\text{sep}}$, and $\mathcal{L}_{\text{sparse}}$; no DINOv2-derived factor targets are consumed.
FactorJEPA	Factorized predictor and top two encoder blocks; approximately 115M trainable parameters.	Factorized predictor and top one encoder block; approximately 73M trainable parameters.	Uses the complete objective: $\mathcal{L}_{\text{JEPA}} + \lambda_{\text{sup}} \mathcal{L}_{\text{factor}} + \lambda_V \mathcal{L}_V + \lambda_{\text{sep}} \mathcal{L}_{\text{sep}} + \lambda_{\text{sparse}} \mathcal{L}_{\text{sparse}}$.
Budget tolerance	LoRA, DoRA, Auto-RGN, and FactorJEPA remain within 5% of the target trainable-parameter budget after exact implementation-level counting.	Same.	If exact instantiated counts exceed the tolerance, LoRA/DoRA rank or the Auto-RGN block budget must be adjusted before training.
Panel E: Resource envelope and profiling contract			
Training hardware	8 × NVIDIA H100 SXM, 80 GiB per GPU.	Same.	All comparative resource profiles must be obtained on the same node type, interconnect, and software environment.
Software environment	PyTorch 2.6.0; CUDA 12.4; cuDNN 9.1; BF16; SDPA enabled.	Same.	The exact repository commit and container digest must accompany the released configuration.
Teacher-cache envelope	At most 42 GPU-hours; peak allocation 48 GiB/GPU.	The same cache is reused; no second scale-specific teacher pass.	Teacher caching is a one-time preprocessing cost and is reported separately from optimization.
Frozen evaluation envelope	No training GPU-hours; inference allocation at most 38 GiB/GPU.	No training GPU-hours; inference allocation at most 30 GiB/GPU.	Zero training cost does not imply zero evaluation cost.
Full fine-tuning envelope	At most 160 GPU-hours; peak allocation at most 78 GiB/GPU.	At most 96 GPU-hours; peak allocation at most 64 GiB/GPU.	Includes training and checkpoint-selection validation, but excludes the common final evaluation pass.
LoRA envelope	At most 96 GPU-hours; peak allocation at most 54 GiB/GPU.	At most 56 GPU-hours; peak allocation at most 44 GiB/GPU.	Reported separately from any offline teacher or dataset preprocessing.
DoRA envelope	At most 100 GPU-hours; peak allocation at most 56 GiB/GPU.	At most 60 GPU-hours; peak allocation at most 46 GiB/GPU.	Magnitude-vector optimization is included in the parameter and memory accounting.
Auto-RGN envelope	At most 104 GPU-hours; peak allocation at most 60 GiB/GPU.	At most 60 GPU-hours; peak allocation at most 48 GiB/GPU.	Includes the fixed gradient-probe pass used for block selection.
FactorJEPA-RAW envelope	At most 112 GPU-hours; peak allocation at most 62 GiB/GPU.	At most 64 GPU-hours; peak allocation at most 50 GiB/GPU.	No offline DINOv2 target-cache cost is attributed to this control.
FactorJEPA envelope	At most 120 optimization GPU-hours; peak allocation at most 64 GiB/GPU.	At most 70 optimization GPU-hours; peak allocation at most 52 GiB/GPU.	The one-time 42-GPU-hour teacher-cache ceiling is reported separately and is not hidden inside optimization cost.

Continued on the next page

Table 6 continued: Canonical FactorJEPA configuration

Configuration item	V-JEPA 2.1 ViT-G	V-JEPA 2.1 ViT-g	Control, interpretation, or measurement rule
GPU-hour definition	$H_{GPU} = \sum_r n_{GPU}^{(r)} \Delta t_r$, where Δt_r is the wall-clock duration of run r .		Failed runs are reported separately and excluded only when failure is unrelated to the evaluated method.
Peak-memory definition	Maximum value returned by <code>torch.cuda.max_memory_allocated()</code> after initialization and five warm-up updates.		Measured using the reported per-device batch, precision, and gradient accumulation.
Latency protocol	Batch 1; 50 warm-up iterations; 500 synchronized timed iterations.	Same.	Report median and 95th-percentile ms/clip. Model-only latency uses resident tensors; end-to-end latency additionally includes decode, resize, normalization, and host-to-device transfer.
Latency ceiling	Model-only median at most 230 ms/clip.	Model-only median at most 160 ms/clip.	These are execution ceilings, not measured results. Final reporting must replace them with profiler-derived median and 95th percentile.
Resource aggregation	Mean and standard deviation across the three training replications.	Same.	GPU-hours are summed per run; peak memory is the maximum per run; latency is summarized from the synchronized inference distribution.

Attribution and Executed Component Ablations

The main results evaluate FactorJEPA as a complete predictive system. This appendix uses the executed comparison family to distinguish three sources of performance: **matched monolithic adaptation**, the **factorized structural package**, and **DINOv2-derived structured supervision**. We then evaluate the executed interaction-objective and encoder-adaptation variants.

The available experiments do not form a complete architecture-supervision factorial. We therefore restrict the attribution language to contrasts directly supported by the executed runs. In particular, when FactorJEPA-RAW includes separation and interaction-sparsity regularization, its comparison with a monolithic predictor measures the contribution of the *factorized structural package*, not predictor architecture in isolation.

Executed Comparison Contract

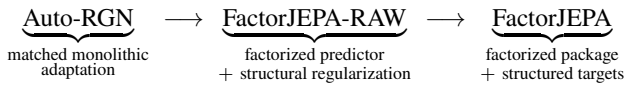
Table 7 defines the training intervention associated with each primary comparison. All trainable methods use the same raw clips, context and target masks, target indices, optimizer, training duration, validation rule, and frozen evaluation protocol. The momentum target encoder remains stop-gradient and follows the same EMA update schedule.

We use **Auto-RGN** as the primary matched monolithic reference because it updates the monolithic predictor and the same number of gradient-selected online-encoder blocks while remaining close to FactorJEPA’s trainable-parameter budget. LoRA and DoRA provide additional parameter-efficient controls. Full fine-tuning is retained only as a non-matched capacity ceiling.

All structured targets are produced automatically by the frozen teacher pipeline. No manually annotated factor labels enter any training condition. The headline evaluators remain frozen and do not reuse DINOv2-derived targets as evaluation labels.

Factorized Structural Package and Structured Targets

The primary attribution chain is



For diagnostic m , let $Y_M^{(m)}$ denote the score of method M . We direction-normalize the metrics as

$$\tilde{Y}_M^{(m)} = \begin{cases} -Y_M^{(m)}, & \text{if lower is better,} \\ Y_M^{(m)}, & \text{if higher is better.} \end{cases}$$

We report three prespecified effects:

$$\begin{aligned} \Delta_{\text{pkg}} &= \tilde{Y}_{\text{FactorJEPA-RAW}} - \tilde{Y}_{\text{Auto-RGN}}, \\ \Delta_{\text{sup|pkg}} &= \tilde{Y}_{\text{FactorJEPA}} - \tilde{Y}_{\text{FactorJEPA-RAW}}, \\ \Delta_{\text{total}} &= \tilde{Y}_{\text{FactorJEPA}} - \tilde{Y}_{\text{Auto-RGN}}. \end{aligned}$$

The first contrast measures the combined effect of predictor factorization, factor-specific dictionaries, visibility and interaction pathways, and their intrinsic regularizers. It is therefore a **package effect**, not a pure architecture effect.

The second contrast is more specific. FactorJEPA and FactorJEPA-RAW share the predictor, synthesis dictionaries, separation and sparsity terms, encoder scope, and optimization protocol. Their difference therefore isolates the incremental contribution of $\mathcal{L}_{\text{factor}}$ and $\mathcal{L}_V$, conditional on the shared factorized structural package.

Figure 17 reports the absolute performance and paired contrasts.

Confidence intervals are estimated using a **paired hierarchical bootstrap**. Each replicate applies the same resampled seeds, cities, source videos, and clips to all methods in a contrast. Cities are resampled first and source videos are resampled within city; all clips from a sampled source video remain together. This pairing removes variation shared by the compared methods and preserves the geographic and temporal dependence of the evaluation set.

Causal L1 is treated as an operational measure of consistency under the frozen evaluator-side intervention protocol, not as evidence of causal identification. Its intervention generator is fixed before model comparison, consumes neither DINOv2 factor targets nor manual factor annotations, and is applied identically to every method.

Factor-Objective and Intervention Variants

The primary-scale ablations contain three configurations of the factorized model:

- **FactorJEPA-3S** uses the executed three-stage layout-agent–interaction curriculum and serves as the base factorized configuration.
- **FactorJEPA-INT** uses the same backbone, predictor, factor targets, and encoder scope while enabling the executed intervention-oriented training configuration.
- **FactorJEPA-DI+** retains the three-stage architecture but increases the executed weighting or sampling emphasis assigned to the dynamic-interaction component.

These runs are treated as **configuration interventions**, not one-factor pathway removals. In particular, FactorJEPA-DI+ tests sensitivity to interaction emphasis; it does not establish that the interaction pathway is necessary. Similarly, FactorJEPA-INT measures the effect of the executed intervention-oriented configuration and should not be interpreted as a general causal-supervision ablation.

Relative to FactorJEPA-3S, define

$$\begin{aligned} \Delta_{\text{INT}} &= \tilde{Y}_{\text{FactorJEPA-INT}} - \tilde{Y}_{\text{FactorJEPA-3S}}, \\ \Delta_{\text{DI+}} &= \tilde{Y}_{\text{FactorJEPA-DI+}} - \tilde{Y}_{\text{FactorJEPA-3S}}. \end{aligned}$$

We evaluate these effects jointly across future prediction, intervention consistency, masking robustness, and motion accessibility. A favorable result on one diagnostic is not treated as a universal improvement when accompanied by degradation on another. This is particularly important for the observed prediction–motion trade-off.

Table 7: **Executed attribution contract.** FactorJEPA-RAW and FactorJEPA share the factorized predictor, separation objective, interaction-sparsity objective, trainable encoder scope, and parameter budget. They differ in access to the DINOv2-derived factor and visibility targets. Auto-RGN is the primary parameter-matched monolithic reference; full fine-tuning is a non-matched capacity ceiling.

Method	Predictor	Factor targets	Executed objective	Trainable scope	Attribution role
Frozen V-JEPA	Monolithic	None	No optimization	None	No-adaptation reference.
V-JEPA Full-FT	Monolithic	None	$\mathcal{L}_{JEPA}$	Complete online encoder and predictor	Non-matched capacity ceiling.
V-JEPA LoRA	Monolithic	None	$\mathcal{L}_{JEPA}$	Matched low-rank projection updates	Parameter-efficient adaptation control.
V-JEPA DoRA	Monolithic	None	$\mathcal{L}_{JEPA}$	Matched decomposed low-rank updates	Parameter-efficient adaptation control.
V-JEPA Auto-RGN	Monolithic	None	$\mathcal{L}_{JEPA}$	Monolithic predictor and selected top- K blocks	Primary matched monolithic reference.
FactorJEPA-RAW	Factorized	None	$\mathcal{L}_{JEPA} + \lambda_{sep}\mathcal{L}_{sep} + \lambda_{sparse}\mathcal{L}_{sparse}$	Factorized predictor and selected top- K blocks	Factorized structural package without teacher targets.
FactorJEPA	Factorized	T_L, T_A, T_V, T_I	$\mathcal{L}_{JEPA} + \lambda_{sup}\mathcal{L}_{factor} + \lambda_V\mathcal{L}_V + \lambda_{sep}\mathcal{L}_{sep} + \lambda_{sparse}\mathcal{L}_{sparse}$	Factorized predictor and selected top- K blocks	Complete model: structural package plus structured supervision.

Parameter matching. At ViT-G, Auto-RGN and FactorJEPA contain approximately 111M and 115M trainable parameters, respectively; at ViT-g, they contain approximately 68M and 73M. Full fine-tuning contains approximately 2.04B and 1.04B trainable parameters and is therefore not used for component attribution.

The final manuscript-facing configuration is selected exclusively by the prespecified validation rule. Test performance is not used to choose among FactorJEPA-3S, FactorJEPA-INT, and FactorJEPA-DI+.

Encoder-Adaptation and Resource Sensitivity

The remaining variants test whether the observed gains can be explained by the amount of encoder adaptation rather than the factorized predictor. V-JEPA LP-FT and full fine-tuning provide the minimal and maximal monolithic adaptation endpoints. The three FactorJEPA-WiseFT variants vary the executed encoder-scope parameter while retaining the factorized predictor and structured objective.

Let $b \in \{30, 50, 70\}$ denote the executed WiseFT scope setting and let P_b , H_b , and M_b denote its trainable parameters, measured GPU-hours, and peak allocated memory. For each setting, we report

$$\mathcal{R}_b = (P_b, H_b, M_b, Y_b^{\text{future}}, Y_b^{\text{causal}}, Y_b^{\text{mask}}, Y_b^{\text{motion}}).$$

The reported setting names must be accompanied by their operational meaning: the exact trainable blocks, frozen blocks, and trainable parameter count. Internal labels such as $\mathbb{f}30$, $\mathbb{f}50$, and $\mathbb{f}70$ are not used without this mapping.

A configuration b is Pareto dominated when another configuration b' satisfies

$$P_{b'} \leq P_b, \quad H_{b'} \leq H_b, \quad \tilde{Y}_{b'}^{(m)} \geq \tilde{Y}_b^{(m)} \quad \forall m \in \mathcal{M}_{\text{primary}}$$

with at least one strict inequality. This analysis distinguishes gains that persist under constrained encoder adaptation from gains obtained solely through additional trainable capacity.

Consolidated Attribution

Figure 17 consolidates the prespecified paired contrasts. Each point reports an absolute, direction-normalized effect; horizontal bars report paired source-clustered 95% confidence intervals. Positive values favor the named intervention.

The evidence is interpreted under the following claim boundaries:

- A positive Δ_{pkg} supports the **factorized structural package**. It does not isolate predictor architecture from separation and sparsity regularization.
- A positive $\Delta_{\text{sup|pkg}}$ supports the incremental contribution of the DINOv2-derived factor and visibility targets, conditional on the shared factorized architecture.
- A positive Δ_{total} establishes improvement over the matched monolithic Auto-RGN reference. Full fine-tuning is interpreted separately because it is not parameter matched.
- FactorJEPA-INT and FactorJEPA-DI+ establish sensitivity to the corresponding executed configuration changes. They do not isolate unexecuted pathway removals.

- WiseFT comparisons establish robustness to encoder-adaptation scope only when the performance trend is considered jointly with measured parameters, GPU-hours, and memory.
- Teacher-relative factor diagnostics may explain a mechanism, but they do not replace the frozen future-latent, intervention, masking, motion, or RGB evaluators.

Accordingly, the executed experiments identify two principal effects: the advantage of the complete factorized structural package over matched monolithic adaptation, and the incremental value of structured teacher targets within that package. Finer architecture-supervision interaction claims are left open because the corresponding supervision-matched monolithic control was not executed.

Metric Definitions, Clustered Inference, and Robustness

This appendix specifies the evaluation contract underlying every reported result. We distinguish **training targets** from **evaluation signals**: DINOv2-derived layout, agent, visibility, and interaction targets supervise FactorJEPa but are not reused as labels for the headline metrics. Unless stated otherwise, every evaluator is frozen before test evaluation and is applied identically to all methods.

Metrics are first computed at their lowest valid evaluation unit and then aggregated through the city-source-video hierarchy. Clips from the same recording are not treated as independent observations. Primary conclusions are restricted to Future-frame MSE, Intervention L1, Mask-ratio slope, and Motion cosine. All remaining diagnostics are treated as secondary or exploratory.

Evaluation Contract and Metric Registry

Table 8 defines the complete metric registry. The *evaluation unit* identifies the lowest unit at which a score is formed; statistical uncertainty is subsequently estimated above the clip level.

Primary Predictive Diagnostics

Future-frame MSE. For clip n , horizon $h \in \mathcal{H}$, and target token $p \in \{1, \dots, m_h\}$, let $\hat{Y}_{n,h,p} \in \mathbb{R}^d$ denote the predicted future token and $Y_{n,h,p}^*$ the stop-gradient target-encoder token. The clip-level error is

$$e_n^{\text{future}} = \frac{1}{|\mathcal{H}|} \sum_{h \in \mathcal{H}} \frac{1}{m_h d} \sum_{p=1}^{m_h} \left\| \hat{Y}_{n,h,p} - Y_{n,h,p}^* \right\|_2^2.$$

All valid target tokens receive equal weight. Target-token normalization, target horizons, and masking are fixed across methods. The executed scorecard reports this MSE quantity; it is not relabeled as L1. Horizon-specific errors are reported as sensitivity analyses and do not replace the pooled primary score.

Intervention L1. Let $\mathcal{G}_{\text{int}}$ be the frozen automatic evaluator-side intervention generator and let

$$\mathcal{A}_n = \mathcal{G}_{\text{int}}(x_n)$$

denote its retained intervention set for clip n . The generator, intervention types, spatial and temporal support rules, and exclusion criteria are frozen before model comparison. It consumes neither DINOv2 factor targets nor manually annotated factor labels.

For intervention $a \in \mathcal{A}_n$, apply the same transformation I_a to the evaluated clip and define

$$\begin{aligned} \Delta \hat{Y}_{n,a} &= \hat{Y}(I_a(x_n)) - \hat{Y}(x_n), \\ \Delta Y_{n,a}^* &= \text{sg}[Y^*(I_a(x_n)) - Y^*(x_n)]. \end{aligned}$$

The normalized intervention error is

$$e_{n,a}^{\text{int}} = \frac{\left\| \Delta \hat{Y}_{n,a} - \Delta Y_{n,a}^* \right\|_1}{\epsilon_{\text{int}} + \left\| \Delta Y_{n,a}^* \right\|_1}.$$

The clip score averages over retained interventions:

$$e_n^{\text{int}} = \frac{1}{\max\{1, |\mathcal{A}_n|\}} \sum_{a \in \mathcal{A}_n} e_{n,a}^{\text{int}}.$$

We report the numerator, target-effect denominator, retained intervention count, and near-zero-effect rate alongside the normalized score. The primary specification uses a fixed ϵ_{int} . Alternative constants and minimum-effect filters are evaluated only as sensitivity checks. This metric measures consistency with intervention-induced latent changes; it does not establish causal identification.

Mask-ratio slope. Let $\mathcal{R} = \{r_1, \dots, r_J\}$ be the prespecified context-mask ratios. For each ratio r , we sample K masks from a fixed sampler and reuse the same realizations across methods. Define

$$e_n(r) = \frac{1}{K} \sum_{k=1}^K e_n^{\text{future}}(M_{r,k} \odot x_n).$$

The primary robustness statistic is the absolute OLS slope

$$\beta_n^{\text{mask}} = \frac{\sum_{r \in \mathcal{R}} (r - \bar{r}) (e_n(r) - \bar{e}_n)}{\sum_{r \in \mathcal{R}} (r - \bar{r})^2},$$

where $\bar{e}_n$ is the mean error over the evaluated ratios. Smaller values indicate slower degradation as context evidence is removed. We also report the complete masking curve,

$$\mathcal{C}_n^{\text{mask}} = \{(r, e_n(r)) : r \in \mathcal{R}\},$$

together with relative slope and masking AUC as secondary robustness summaries.

Table 8: **Evaluation metric registry.** The table records the operational definition, preferred direction, evaluation unit, frozen signal or evaluator, and inferential status of every reported diagnostic. “Intervention L1” corresponds to the quantity previously labeled Causal L1; the revised name reflects that the metric measures intervention-response consistency rather than causal identification.

Metric	Operational definition	Dir.	Unit	Frozen signal or evaluator	Status
Primary predictive diagnostics					
Future-frame MSE	Mean squared error between predicted and target future-token embeddings.	↓	Clip–horizon	Momentum target encoder	Primary
Intervention L1	Normalized L1 discrepancy between predicted and target latent changes under the same automatically generated intervention.	↓	Clip–intervention	Intervention generator and target encoder	Primary
Mask-ratio slope	OLS slope of Future-frame MSE as the visible context is progressively reduced.	↓	Clip–mask curve	Target encoder and fixed mask sampler	Primary
Motion cosine	Cosine similarity between linearly decoded and target motion descriptors.	↑	Clip	Motion estimator and fixed-capacity probe	Primary
Semantic diagnostics					
Action top-1	Top-1 accuracy of a fixed-capacity action probe.	↑	Clip	Frozen action labels and probe protocol	Secondary
Taxonomy F1	F1 over the shared dynamic-agent taxonomy.	↑	Clip–class	Frozen taxonomy evaluator	Secondary
Prediction-stability diagnostics					
Rollout-drift slope	Increase in prediction error over autoregressive rollout depth.	↓	Clip–rollout	Target encoder	Secondary
L1-vs- Δt decay	Increase in latent error over the evaluated future horizons.	↓	Clip–horizon	Target encoder	Secondary
Exposure-bias gap	Difference between free-running and teacher-conditioned prediction error.	↓	Clip	Target encoder	Secondary
Temporal diagnostics					
Frame-order sensitivity	Prediction-error increase after controlled frame-order corruption.	↑	Clip	Fixed temporal permutation	Secondary
Arrow-of-Time	Accuracy for distinguishing forward from reversed clips.	↑	Clip	Fixed temporal classifier	Secondary
Temporal-order accuracy	Accuracy for detecting frame permutations.	↑	Clip	Fixed temporal-order classifier	Secondary
Playback-pace accuracy	Accuracy for identifying the applied temporal-rate transformation.	↑	Clip	Fixed pace classifier	Secondary
TCC cycle-back error	Temporal distance between a source frame and its cycle-consistent match.	↓	Frame pair	Frozen correspondence features	Secondary
TCC Kendall τ	Rank agreement between predicted and true temporal correspondences.	↑	Clip pair	Frozen correspondence features	Secondary

Motion cosine. Let E_{mot} be the independently frozen motion estimator and

$$u_n^* = E_{\text{mot}}(x_{n,t:t+\Delta}) \in \mathbb{R}^{d_{\text{mot}}}$$

its target descriptor. For every world model M , a fixed-capacity linear probe W_M is fitted on the training split:

$$W_M = \arg \min_W \sum_{n \in \mathcal{D}_{\text{train}}} \left\| W \text{ pool } \left(\hat{Y}_n^M \right) - u_n^* \right\|_2^2 + \lambda_{\text{mot}} \|W\|_F^2.$$

The regularization coefficient is selected on the validation split under the same grid for every method. The test descriptor is

$$\hat{u}_n^M = W_M \text{ pool } \left(\hat{Y}_n^M \right),$$

and Motion cosine is

$$s_n^{\text{motion}} = \frac{\langle \hat{u}_n^M, u_n^* \rangle}{\|\hat{u}_n^M\|_2 \|u_n^*\|_2 + \epsilon_{\text{mot}}}.$$

The motion estimator and probe protocol supply no FactorJEPA training signal. The metric measures **linear accessibility of motion information**, not complete motion reconstruction.

Secondary Semantic and Temporal Diagnostics

Semantic probes. Action top-1 and Taxonomy F1 use fixed-capacity probes trained on the training split and selected on the validation split. Probe architecture, optimization, and regularization are identical across world models. Taxonomy F1 uses the same class support and averaging convention for every method; unsupported classes are not silently removed on a method-specific basis.

Rollout and horizon stability. Let $e_n^{(q)}$ denote future-token error after rollout step q . Rollout-drift slope is the OLS coefficient of $e_n^{(q)}$ against q . Similarly, L1-vs- Δt decay is the slope of latent prediction error over the evaluated future horizons. Both are computed using a fixed set of rollout depths and horizons.

The free-running exposure-bias gap is

$$g_n^{\text{exp}} = e_{n,\text{free}}^{\text{future}} - e_{n,\text{teacher}}^{\text{future}},$$

where both errors use the same target encoder and future horizon.

Order and pace sensitivity. Frame-order sensitivity is the increase in prediction error after a fixed temporal permutation:

$$s_n^{\text{order}} = e_n^{\text{future}}(I_{\text{perm}}(x_n)) - e_n^{\text{future}}(x_n).$$

Arrow-of-Time, temporal-order, and playback-pace accuracy use frozen evaluation protocols with fixed transformation classes. Arrow-of-Time distinguishes forward from reversed clips; temporal-order detects frame permutations; playback-pace identifies the applied temporal-rate transformation.

Temporal correspondence. For each source frame, the temporal-correspondence evaluator identifies a nearest match in the paired sequence and maps that match back to the source. TCC cycle-back error measures the resulting temporal distance. TCC Kendall τ measures rank agreement between predicted and ground-truth temporal ordering. Both use the same frozen features and matching rule for every method.

Paired Clustered Inference and Multiplicity

Let $y_{M,s,c,v,i}$ denote a metric for method M , training seed s , city c , source video v , and clip i . For metrics with multiple horizons, masks, or interventions, those observations are first reduced to one clip-level score using the definitions above.

Primary estimand. To prevent high-volume cities or recordings from dominating the evaluation, we use equal weighting at the city and source-video levels:

$$\bar{y}_{M,c,v} = \frac{1}{S} \sum_{s=1}^S \frac{1}{|\mathcal{I}_{c,v}|} \sum_{i \in \mathcal{I}_{c,v}} y_{M,s,c,v,i},$$

$$\hat{\mu}_M = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \frac{1}{|\mathcal{V}_c|} \sum_{v \in \mathcal{V}_c} \bar{y}_{M,c,v}.$$

The primary paired effect between methods A and B is

$$\hat{\delta}_{A,B} = \tilde{\mu}_A - \tilde{\mu}_B,$$

where the direction-normalized mean is

$$\tilde{\mu}_M = \begin{cases} -\hat{\mu}_M, & \text{if lower is better,} \\ \hat{\mu}_M, & \text{if higher is better.} \end{cases}$$

Consequently, $\hat{\delta}_{A,B} > 0$ always favors method A . Clip-weighted estimates are reported as an aggregation sensitivity, not as the primary estimand.

Paired hierarchical bootstrap. For each bootstrap replicate, we:

1. resample cities with replacement;
2. resample source videos within each sampled city;
3. retain all clips from every sampled source video;
4. apply the same sampled cities, videos, clips, masks, horizons, and interventions to all compared methods; and
5. recompute the complete metric and aggregation pipeline.

This preserves within-recording dependence and method pairing. We report 95% BCa intervals for absolute method means and paired effects. The source-video group, rather than the clip, is the lowest independently resampled unit.

Training-seed variation. For every method and metric, we report all $S = 3$ executed seed values, their mean, and

$$\text{SD}_{\text{seed}}(M) = \sqrt{\frac{1}{S-1} \sum_{s=1}^S (\hat{\mu}_{M,s} - \bar{\mu}_M)^2}.$$

The primary geographic estimand averages the three seeds before city-level aggregation. As a sensitivity analysis, a crossed bootstrap resamples seeds independently of cities while applying the same resampled seed indices to all methods. Because only three seeds are available, seed is not treated as a primary random-effect variance component.

Mixed-effects sensitivity. As a model-based check, we fit

$$y_{M,s,c,v,i} = \beta_0 + \beta_M^{\text{method}} + \gamma_s^{\text{seed}} + u_c^{\text{city}} + u_{v(c)}^{\text{source}} + \varepsilon_{M,s,c,v,i},$$

where method is the effect of interest, seed is a fixed blocking factor, and city and source video are nested random intercepts. The mixed-effects analysis is considered supportive only when its effect direction agrees with the paired clustered estimate.

Multiple comparisons. Within each backbone, the primary confirmatory family contains the prespecified attribution contrasts crossed with the four primary diagnostics. Two-sided paired-bootstrap p -values are adjusted using Holm's step-down procedure. The secondary scorecard is controlled using Benjamini-Hochberg FDR and is explicitly labeled exploratory.

We report unadjusted absolute effects and clustered intervals together with adjusted decisions. An unadjusted 95% interval excluding zero is not described as Holm-corrected evidence unless the corresponding adjusted decision also passes the prespecified level.

Metric Sensitivity and Complete Numerical Results

Robustness is evaluated along four prespecified axes.

Metric-construction sensitivity. For future prediction, we report horizon-specific MSE and, where executed, tokenwise L1 as a secondary alternative. For intervention consistency, we vary ϵ_{int} and the minimum target effect while reporting the retained intervention fraction. For missing-evidence

robustness, we compare absolute slope, relative slope, and masking AUC while preserving the same mask realizations.

A conclusion is considered stable only when its direction is preserved across the prespecified variants. The primary metric definition is not changed after inspecting test performance.

Aggregation sensitivity. We compare:

- equal-city and clip-weighted means;
- city–source-video and source-video-only bootstrap intervals;
- seed-averaged and crossed seed-resampling estimates; and
- complete and leave-one-city-out evaluation.

For each leave-one-city-out run, all clips from the omitted city are removed before recomputing the metric and method contrast.

Cross-scale ranking robustness. For diagnostic m , let $\mathbf{s}_G^{(m)}$ and $\mathbf{s}_g^{(m)}$ contain the direction-normalized scores of the common method set at the ViT-G and ViT-g scales. We report

$$\rho_m = \text{Spearman}(\mathbf{s}_G^{(m)}, \mathbf{s}_g^{(m)}),$$

$$\tau_m = \text{Kendall}(\mathbf{s}_G^{(m)}, \mathbf{s}_g^{(m)}).$$

Because the evaluated methods are a fixed comparison set, these correlations are interpreted descriptively. Robustness is assessed through leave-one-method-out and leave-one-family-out ranges:

$$\mathcal{R}_{\rho,m} = \left[\min_j \rho_m^{(-j)}, \max_j \rho_m^{(-j)} \right].$$

Method families comprise frozen, conventional fine-tuning, parameter-efficient adaptation, FactorJEPA variants, and encoder-scope variants. Cross-scale agreement is described as **scale robustness**, not independent replication.

Complete scorecards. Figures 17 and 17 show all executed diagnostics for the ViT-G and ViT-g backbones. Exact primary values and clustered intervals are reported in Figure 17. The complete numerical scorecard additionally records, for every method and diagnostic:

(mean, 95% clustered CI, seed SD, paired reference effect, adjusted decision)

The graphical scorecards provide the broad performance profile, whereas the numerical tables and prespecified contrasts determine the inferential conclusions. Accordingly, no method is declared uniformly superior on the basis of isolated secondary wins.

Latent-to-RGB Decoding and Qualitative Analysis

The latent-to-RGB decoder provides a **secondary, interpretability-oriented evaluation** of the predicted future representation. It does not contribute to FactorJEPA training

and is not used for model selection. Its purpose is to expose what a predicted future latent implies in pixel space under a fixed rendering model.

The evaluation follows three constraints. First, FactorJEPA and all comparison models remain frozen during decoder training. Second, one decoder is trained per backbone scale using only target-encoder latents from the training partition. Third, the selected decoder is frozen and applied without method-specific adaptation to every predicted-latent source. Consequently, RGB differences within a backbone scale cannot be attributed to different decoder capacities or optimization budgets.

Decoder Architecture and Latent Transport

Let $Y \in \mathbb{R}^{B \times T_y \times P \times d_J}$ denote a sequence of future JEPA tokens, where $P = h_J w_J$ is the spatial token count and $d_J \in \{1664, 1408\}$ for the ViT-G and ViT-g backbones, respectively. The Cosmos decoder expects a spatially organized conditioning tensor rather than an unordered JEPA token sequence. We therefore introduce a trainable transport operator $\mathcal{T}_\omega$ that aligns token dimensionality, spatial organization, temporal order, and feature statistics.

For future index τ , the transported representation is

$$U_\tau = \text{reshape}_{h_J \times w_J} [W_{\text{in}} \text{LN}(Y_\tau)],$$

$$Z_\tau = \mathcal{B}_\omega(U_\tau + E_{\text{space}} + E_{\text{time}}(\tau)),$$

$$\mathcal{T}_\omega(Y)_\tau = W_{\text{out}} \text{LN}(Z_\tau) \in \mathbb{R}^{h_C \times w_C \times d_C},$$

where W_{in} and W_{out} are learned projections, $\mathcal{B}_\omega$ is the executed transport stack, and E_{space} and E_{time} preserve spatial and temporal position. Any interpolation required to obtain $h_C \times w_C$ is performed inside $\mathcal{T}_\omega$ and is identical for all latent sources at a fixed backbone scale.

The rendered future is

$$\hat{x} = \mathcal{D}_\Omega(Y) = \mathcal{G}_{\psi_C}(\mathcal{T}_\omega(Y)), \quad \Omega = \{\omega, \psi_C^{\text{adapt}}\},$$

where $\mathcal{G}_{\psi_C}$ is initialized from the reported Cosmos checkpoint (NVIDIA 2025). The notation ψ_C^{adapt} denotes exactly those Cosmos parameters updated during decoder adaptation; all remaining Cosmos parameters stay frozen. After validation-based checkpoint selection, the complete mapping $\mathcal{D}_\Omega$ is frozen.

Decoder Training and Comparison Neutrality

For a training example n , let x_n^+ denote the observed future RGB target and let

$$Y_n^* = \text{sg}[\bar{f}_{\hat{\theta}}(x_n^+)]$$

be the corresponding frozen target-encoder latent. The decoder is trained **only** on pairs $\{(Y_n^*, x_n^+)\}$ from the training partition:

$$\Omega^* = \arg \min_{\Omega} \frac{1}{|\mathcal{D}_{\text{train}}|} \sum_{n \in \mathcal{D}_{\text{train}}} \mathcal{L}_{\text{decode}}(\mathcal{D}_\Omega(Y_n^*), x_n^+).$$

Predicted latents from FactorJEPA, FactorJEPA-RAW, or any adaptation baseline are excluded from decoder optimization and checkpoint selection. This prevents the decoder from

Table 9: **Latent-to-RGB decoder contract.** One scale-specific decoder is trained for each JEPA embedding width. Within a backbone scale, every method uses exactly the same transport operator, Cosmos component, preprocessing, and evaluation settings. Fields marked *from log* must be copied verbatim from the executed decoder configuration.

Component	Executed configuration	Comparison control
JEPA latent source	ViT-G: $d_J = 1664$; ViT-g: $d_J = 1408$; future frames 13:16 from the common 384×384 , 16-frame protocol.	Target and predicted latents use the same temporal indices, normalization, token ordering, and spatial support.
Transport operator	Layer normalization, input projection, spatial reshaping, spatiotemporal transport stack, output projection, and positional encoding. Exact depth, width, heads, and d_C : <i>[from log]</i> .	One transport operator per backbone scale; no method-specific projection or normalization.
Cosmos initialization	Checkpoint identifier and revision: <i>[from log]</i> . Checkpoint hash: <i>[from log]</i> .	The same initialization is used for all methods evaluated at a fixed backbone scale.
Cosmos update scope	Trainable modules: <i>[from log]</i> . Frozen modules: <i>[from log]</i> .	The update scope is fixed before decoder training and is not selected separately for individual prediction methods.
RGB output	Frames, resolution, color range, and frame rate: <i>[from log]</i> .	All outputs undergo the same clipping, resizing, and inverse normalization before evaluation.
Frozen upstream models	Online encoder, momentum target encoder, monolithic or factorized predictor, DINOv2 teacher, and factor-target operators.	RGB reconstruction gradients never enter a world model or pseudo-target generator.
Frozen RGB evaluators	LPIPS network, optical-flow estimator, and agent detector. Exact checkpoints and thresholds: <i>[from log]</i> .	No RGB evaluator supplies FactorJEPA supervision or participates in decoder optimization.

adapting preferentially to the latent distribution of one prediction method.

The reconstruction objective is

$$\mathcal{L}_{\text{decode}} = \lambda_{\text{pix}} \mathcal{L}_{\text{char}} + \lambda_{\text{perc}} \mathcal{L}_{\text{LPIPS}} + \lambda_{\text{str}} \mathcal{L}_{\text{SSIM}} + \lambda_{\text{mot}} \mathcal{L}_{\text{motion}} + \lambda_{\text{reg}} \mathcal{R}(\Omega),$$

where the robust pixel term is

$$\mathcal{L}_{\text{char}} = \frac{1}{|\mathcal{P}|} \sum_{p \in \mathcal{P}} \sqrt{(\tilde{x}(p) - x^+(p))^2 + \epsilon_{\text{char}}^2}.$$

Here, $\mathcal{L}_{\text{LPIPS}}$ measures perceptual discrepancy, $\mathcal{L}_{\text{SSIM}} = 1 - \text{SSIM}$ penalizes structural distortion, and $\mathcal{L}_{\text{motion}}$, when enabled in the executed configuration, compares frozen-estimator flow between the final context frame and the reconstructed or observed future. Terms assigned zero weight are disabled rather than silently omitted.

The checkpoint is selected exclusively by the prespecified validation criterion. No test frame, test latent, test metric, or prediction from a compared method influences decoder training, early stopping, or hyperparameter selection.

At evaluation, the selected parameters Ω^* are frozen and

$$\hat{x}_{n,M} = \mathcal{D}_{\Omega^*}(\hat{Y}_{n,M})$$

is computed for every method M using identical preprocessing, sampling parameters, precision, and random seeds. When decoding is stochastic, all methods use the same pre-specified seed set and results are averaged over that set. Best-of- K sample selection is not used for the primary comparison.

Oracle Reconstruction and Forecast-Gap Decomposition

A decoded prediction can fail because the predicted latent is incorrect, because the decoder cannot invert even a correct

target latent, or because both effects interact. We expose these sources through an **oracle target-latent control**.

For each held-out example, define

$$\tilde{x}_n^* = \mathcal{D}_{\Omega^*}(Y_n^*), \quad \hat{x}_{n,M} = \mathcal{D}_{\Omega^*}(\hat{Y}_{n,M}).$$

For a lower-is-better distortion d , the three reported quantities are

$$\begin{aligned} E_{\text{oracle}} &= d(\tilde{x}_n^*, x_n^+), \\ E_{\text{e2e}}(M) &= d(\hat{x}_{n,M}, x_n^+), \\ G_{\text{visible}}(M) &= d(\hat{x}_{n,M}, \tilde{x}_n^*). \end{aligned}$$

The oracle term measures the decoder’s reconstruction limitation, the end-to-end term measures the complete prediction-and-rendering system, and G_{visible} measures how strongly replacing the target latent with the predicted latent changes the rendered future. For metric distances such as pixel L_1 ,

$$E_{\text{e2e}}(M) \leq G_{\text{visible}}(M) + E_{\text{oracle}}$$

by the triangle inequality. This bound is not asserted for similarity scores or learned perceptual measures that are not guaranteed metrics.

We additionally report the oracle-relative degradation. For a lower-is-better metric $Q^\downarrow$,

$$\Delta_Q^\downarrow(M) = Q^\downarrow(\hat{x}_{n,M}, x_n^+) - Q^\downarrow(\tilde{x}_n^*, x_n^+),$$

whereas for a higher-is-better metric $Q^\uparrow$,

$$\Delta_Q^\uparrow(M) = Q^\uparrow(\tilde{x}_n^*, x_n^+) - Q^\uparrow(\hat{x}_{n,M}, x_n^+).$$

Positive values indicate degradation relative to oracle decoding. These quantities are called **decoder-conditioned forecast deficits**; they are not interpreted as an exact additive decomposition because $\mathcal{D}_{\Omega^*}$ is nonlinear.

Table 10: **Latent-to-RGB evaluation under a common frozen decoder.** Entries should be reported as held-out mean [95% clustered CI]. Values in parentheses denote the direction-normalized deficit relative to target-latent oracle decoding; positive deficits indicate worse performance than the oracle. No predicted-latent source is used to train or select the decoder.

Latent source	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	Flow EPE $\downarrow$	Agent F1 $\uparrow$
Target latent (<i>oracle</i>)	[result]	[result]	[result]	[result]	[result]
V-JEPA Auto-RGN	[result]	[result]	[result]	[result]	[result]
FactorJEPA-RAW	[result]	[result]	[result]	[result]	[result]
FactorJEPA	[result]	[result]	[result]	[result]	[result]

Quantitative Decoded-Output Evaluation

We evaluate every decoded future using the same held-out examples and five complementary diagnostics:

- **PSNR $\uparrow$** measures pixel-level fidelity relative to the observed future;
- **SSIM $\uparrow$** measures preservation of local luminance, contrast, and structural organization;
- **LPIPS $\downarrow$** measures perceptual discrepancy using one fixed pretrained feature network;
- **Flow EPE $\downarrow$** measures motion disagreement between the observed and decoded futures using one frozen optical flow estimator; and
- **Agent F1 $\uparrow$** measures preservation of detectable dynamic agents using one frozen detector and a fixed matching threshold.

For motion evaluation, let x_n^c be the final context frame. Using the frozen estimator Φ_{flow} ,

$$F_n^* = \Phi_{\text{flow}}(x_n^c, x_n^+), \quad \hat{F}_{n,M} = \Phi_{\text{flow}}(x_n^c, \hat{x}_{n,M}),$$

and

$$\text{EPE}_{n,M} = \frac{1}{|\mathcal{P}|} \sum_{p \in \mathcal{P}} \left\| \hat{F}_{n,M}(p) - F_n^*(p) \right\|_2.$$

Agent F1 uses detections produced independently on the observed and decoded future frames. Predictions are matched by the prespecified class-compatible overlap rule; the detector, confidence threshold, matching threshold, and taxonomy are held fixed across methods. This metric assesses *detector-visible agent preservation*, not photorealism or manually verified object correctness.

We further report results over prespecified density and visibility strata. Strata are computed from the fixed automatic measurement pipeline before RGB evaluation and remain unchanged across methods. Boundaries are derived only from the training distribution. Clips within a source video retain the same cluster identity when estimating uncertainty. These analyses test whether decoded-output quality degrades disproportionately in crowded or heavily occluded scenes; they are secondary to the complete held-out comparison.

Qualitative Comparisons and Failure Analysis

Qualitative examples use a fixed column order:

context | observed future | oracle decode
 | V-JEPA | FactorJEPA-RAW | FactorJEPA.

The oracle column is retained in every row because it distinguishes an artifact already present when decoding the correct target latent from an error introduced by future prediction. All methods use the same display range, output resolution, temporal index, decoder checkpoint, and sampling seed.

For an additional functional diagnostic, we remove one factor contribution before decoding:

$$\hat{x}_n^{(-k)} = \mathcal{D}_{\Omega^*} \left(\hat{Y}_n - \hat{Y}_{n,k} \right), \quad k \in \{L, A, I\},$$

and visualize the normalized influence map

$$\mathcal{I}_{n,k}(p) = \frac{\left| \hat{x}_n(p) - \hat{x}_n^{(-k)}(p) \right|}{\epsilon + \sum_{q \in \mathcal{P}} \left| \hat{x}_n(q) - \hat{x}_n^{(-k)}(q) \right|}.$$

These maps show where the rendered output is sensitive to removal of a predictive pathway. They are **functional perturbation diagnostics**, not causal pixel explanations and not evidence of identifiable latent factors.

Failure cases are categorized using observable symptoms and the oracle control:

Interpretation. RGB decoding offers an interpretable view of future-latent predictions, but it remains a decoder-conditioned diagnostic. The latent-space metrics provide the primary comparison because they evaluate predicted representations directly. RGB metrics and qualitative results provide complementary evidence about whether those representations preserve visually recoverable layout, agent occupancy, interaction structure, and motion under one fixed rendering interface.

Table 11: **Operational taxonomy and attribution rules for decoded-future failures.** Let $X = x^+$ denote the observed future, $O = \hat{x}^*$ the target-latent oracle reconstruction, and $P = \hat{x}_M$ a decoded prediction. Panel A uses the matched triplet (X, O, P) to localize error; Panel B defines observable failure modes and corroborating diagnostics. Attribution is diagnostic rather than causal because an off-manifold predicted latent may expose nonlinear decoder sensitivities.

Class	Operational signature	Oracle comparison	Interpretation or diagnostic
Panel A: Oracle-based attribution rules			
Oracle-present	The same artifact is visible in both O and P relative to X.	O already fails to reconstruct the relevant content.	Primarily a decoder-capacity or target-latent-invertibility limitation; it must not be attributed solely to forecasting.
Prediction-conditioned	O preserves the relevant content, whereas P does not.	The artifact appears only after replacing the target latent with $\hat{Y}_M$.	Localized to the prediction-conditioned pathway: inaccurate predicted latent, distribution shift at the decoder input, or their interaction.
Mixed or amplified	The artifact is present in O but becomes materially stronger in P.	Both oracle reconstruction error and oracle-to-prediction deviation are non-negligible.	Decoder and forecasting effects coexist; report oracle error, end-to-end error, and visible forecast gap separately.
Panel B: Observable decoded-future failure modes			
Layout drift	Static boundaries, façades, road geometry, or background support move despite being stable in X.	Check whether the same displacement occurs in O.	Static-region error, boundary displacement, and spurious background flow distinguish geometric drift from local texture variation.
Agent omission	An agent visible in X is absent, severely attenuated, or merged into the background in P.	Determine whether the corresponding agent is recoverable in O.	Agent recall, Agent F1, and dynamic-region error quantify the failure; small or heavily occluded agents should be reported separately.
Agent duplication	One reference agent produces multiple overlapping instances or spatially inconsistent fragments in P.	If duplication is also present in O, classify it as oracle-present.	Agent precision, duplicate-detection rate, and connected-component fragmentation provide corroborating evidence.
Interaction inconsistency	Nearby agents exhibit incorrect relative ordering, separation, direction, or collision geometry.	Compare pairwise geometry in X, O, and P.	Pairwise displacement, relative-motion error, and interaction-edge agreement distinguish interaction failure from independent agent misplacement.
Motion under-dispersion	Moving agents become blurred, nearly stationary, or displaced toward an average future.	An accurate O with reduced motion only in P indicates prediction-conditioned averaging.	Flow EPE, predicted-to-reference flow-magnitude ratio, and dynamic-region sharpness quantify the effect. It may reflect deterministic prediction under multimodal futures.
Appearance substitution	Coarse location and occupancy remain plausible, but local appearance or category-specific structure changes.	Determine whether the substitution is already visible in O.	Agent-crop LPIPS and frozen-descriptor similarity separate appearance loss from geometric or occupancy failure.
Unsupported content	P introduces an agent, boundary, texture, or motion pattern unsupported by X.	Oracle presence suggests decoder hallucination; prediction-only presence indicates an off-manifold latent or prediction-decoder interaction.	False-positive agent rate, static-region residuals, perceptual error, and oracle comparison jointly support classification.

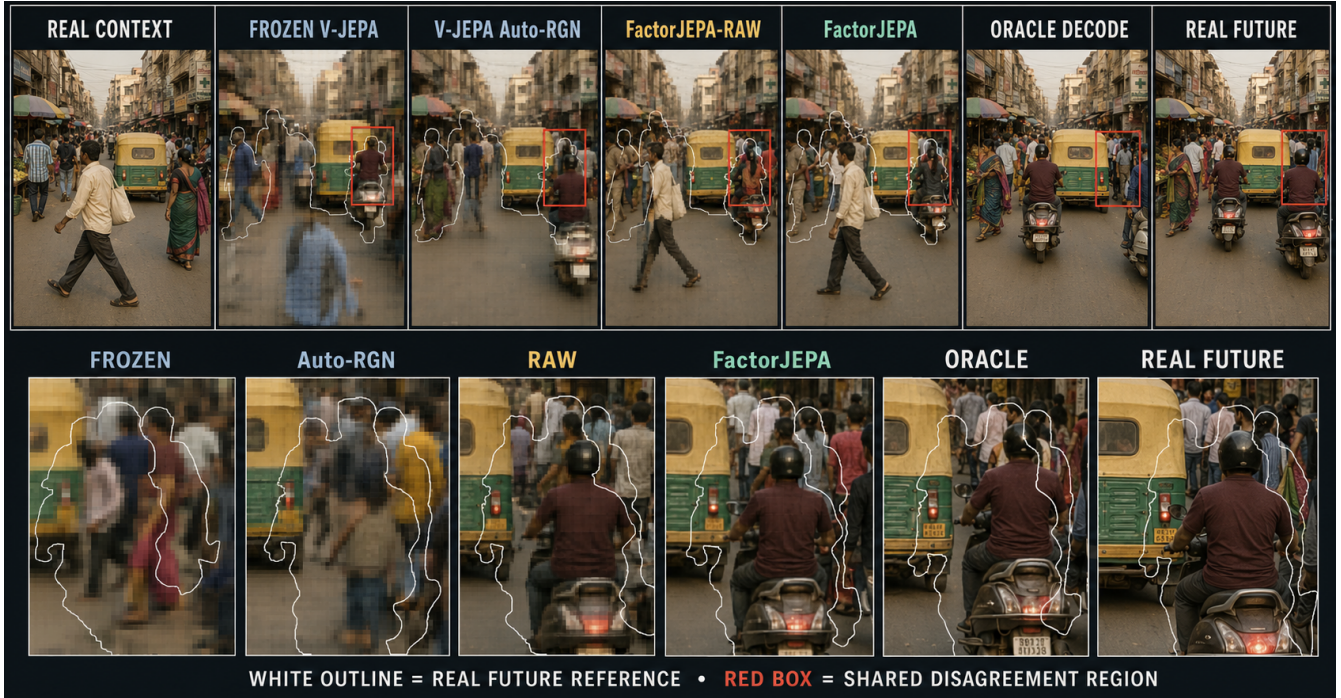


Figure 18: **Illustrative layout for future-consistency inspection.** The top row compares the observed context with decoded futures from frozen V-JEPA, V-JEPA Auto-RGN, FactorJEPA-RAW, and FactorJEPA, followed by the target-latent oracle reconstruction and observed future. The identical white contour, derived from the observed future, is overlaid on every predicted panel to expose disagreement in agent occupancy, position, visibility, and interaction geometry. Red boxes identify a common high-disagreement region, enlarged in the bottom row. The synthesized panels demonstrate the intended visualization protocol and are not experimental model outputs.