SUPPORT×OPERATION FACTORIZATION: COMPOSITIONAL READOUT OF FROZEN VISION ENCODERS UNDER CONTROLLED INTERVENTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Frozen vision encoders encode responses to spatial image edits, yet factor-wise probes do not test whether an operation remains bound to the region where it occurs. They can therefore credit multiple operations through the same predicted slot, a degeneracy we call *operation laundering*. We introduce an injectively aligned leave-one-cell-out evaluation that requires distinct ground-truth axis values to occupy distinct slots. Our Support–Operation Factorization (SO-OPF) readout separates support salience from operation identity, making the binding between *where* and *what* explicit. We distinguish readout capacity under a known factorial assignment from recovery of that assignment using only flat cell labels. Across controlled synthetic scenes and image-disjoint natural images, SO-OPF composes held-out support–operation pairs, while learned assignments retain much of the performance obtained with known structure. Under the same flat-label objective and a shared fixed training budget, SO-OPF outperforms the tested Dense-MLP and Joint-MLP readouts across encoder families and multiple data constructions. Router controls distinguish position-driven from source-driven routing: source features support generalization to unseen layouts, while edit-space transfer characterizes robustness across intervention realizations. These results establish a controlled framework for evaluating readout capacity, assignment recovery, and alignment separately.

1 INTRODUCTION

Compositional generalization asks whether a representation can recombine familiar factors in configurations that were not observed during training. For visual change, the relevant factors include not only *what* operation occurred, but also *where* it was applied. A model that recognizes hue edits on floors and pattern edits on walls may still fail on a hue edit applied to a wall. Factor-wise probes can miss this failure because recovering each attribute separately does not demonstrate that the representation preserves their binding (Montero et al., 2021; Schott et al., 2022).

Frozen vision encoders offer a controlled setting in which to study this question: the representation remains fixed, so success must come from information already present in its token map rather than from task-specific fine-tuning. Recent progress in self-supervised and predictive representation learning (Siméoni et al., 2025; Assran et al., 2023; Bardes et al., 2024) therefore raises a concrete question: do token-level responses to controlled edits support new combinations of operation and spatial support? This question connects disentanglement (Higgins et al., 2017; Kim & Mnih, 2018) with reasoning about change in predictive world models (LeCun, 2022; Garrido et al., 2024). In particular, a representation may encode the type of change while failing to bind it to the region where it occurred.

We study this problem using paired source and intervened images that differ by a controlled visual edit. Their patch-token difference, which we call the *innovation field*, describes how a frozen encoder responds to the intervention. We organize interventions in a support-by-operation grid and hold out one cell at a time (Figure 1). The readout must then recognize the unseen pairing even though both its support and its operation occur elsewhere in the training data. This protocol tests recombination rather than ordinary recognition of either axis.

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

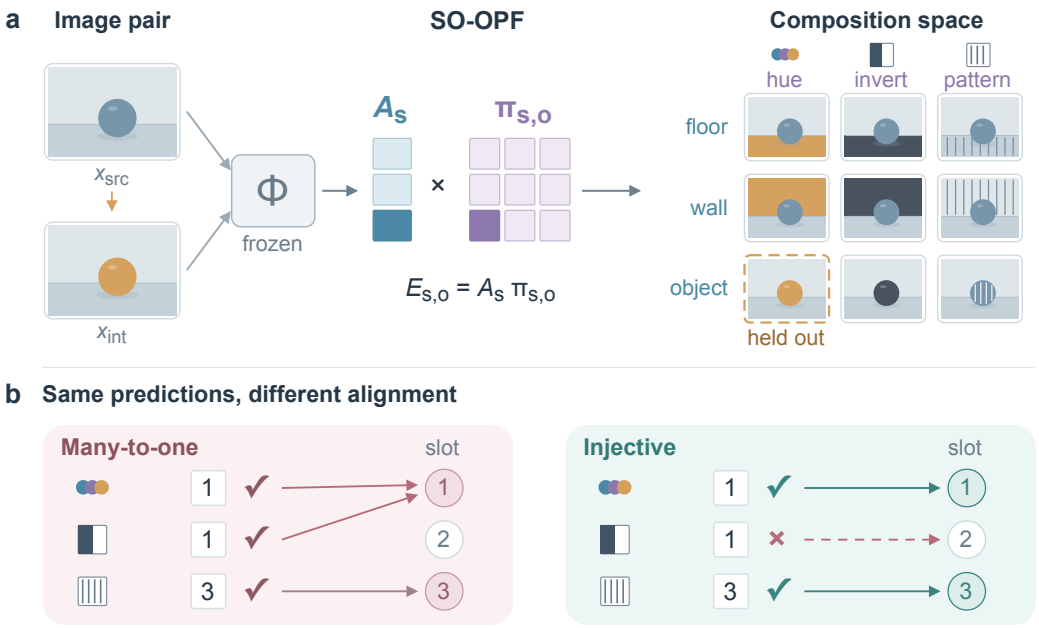

Figure 1: **Compositional readout and injective evaluation. a**, Frozen features yield support salience A_s and conditional operation posteriors $\pi_{s,o}$. The grid depicts intervention classes, with one pairing held out. **b**, Square badges show predicted operation slots, and arrows align ground-truth labels with those slots. Many-to-one alignment credits hue and inversion through slot 1. Injective alignment instead assigns inversion to slot 2, so its slot-1 prediction is counted as incorrect. Checks and crosses indicate agreement with the selected alignment. All images and responses are schematic.

Evaluating such recombination introduces a further difficulty. The internal slots of a readout are permutation-invariant and must be aligned with the ground-truth axes after training. A many-to-one alignment can assign several operations to the same predicted slot and credit all of them, even when the readout has not learned distinct operation identities. We call this degeneracy *operation laundering*. Our evaluation prevents it through an injective assignment, estimated only from training pairs, that requires every ground-truth axis value to occupy a distinct slot.

To investigate what information the frozen representation carries, we introduce Support–Operation Factorization (SO-OPF). The readout separately estimates where the innovation is concentrated and what operation produced it, then combines the two predictions at the cell level. This inductive bias makes the desired binding explicit without altering the encoder. It does not, however, automatically reveal how flat observed-cell labels correspond to locations in the factorial grid.

We therefore separate two questions that are often conflated. The first is *readout capacity*: if the support–operation taxonomy is known, can the readout compose a held-out cell? The second is *assignment recovery*: can the mapping from flat cell labels to the factorial grid be learned without separate axis labels? A compressed label-order control tests whether an arbitrary ordering of the observed labels is sufficient, while an alternative rendering environment tests whether the conclusions transfer across visual environments.

Across synthetic scenes and image-disjoint natural-image interventions, the results show that frozen token representations can support held-out compositions when the factorial assignment is known. Learning the assignment from flat labels preserves much of this ability but introduces a measurable gap, whereas imposing an arbitrary label order fails to recover the grid. Under the same flat-label core objective and fixed training budget, SO-OPF also outperforms matched Dense-MLP and Joint-MLP readouts across multiple data constructions. Position and layout controls identify when routing uses fixed geometry and show that source features support unseen layouts. Together with edit-space transfer tests, these findings characterize both the benefits of readout structure and its behavior across intervention designs.

Our contributions are:

1. We formulate compositional readout over a support-by-operation grid and introduce a leave-one-cell-out metric with injective alignment that detects operation laundering.
2. We introduce SO-OPF, a factorized readout that separates spatial support from operation identity and tests their recombination without fine-tuning the vision encoder, and compare it with matched readouts under the same flat-label objective and fixed training budget.
3. We empirically separate readout capacity from assignment recovery across synthetic, rendered, and natural-image interventions, including controls for label order, position, unseen layouts, edit realization, encoder family, and data construction.

2 Related Work

Factors and compositional generalization. Disentangled representations need inductive bias (Locatello et al., 2019), and recovering individual factors does not guarantee generalization to their unseen combinations (Montero et al., 2021; Schott et al., 2022). Systematic generalization benchmarks (Lake & Baroni, 2018; Keysers et al., 2020) and compositional zero-shot learning (Misra et al., 2017) test this distinction explicitly. Our support–operation grid applies the same principle to paired visual interventions, evaluating a structured readout on a frozen encoder rather than training an end-to-end attribute–object classifier.

Spatial slots and structured probes. Object-centric models such as Slot Attention (Locatello et al., 2020) separate spatial entities; our readout separates operation identity into a second axis. This binding is related to tensor-product representations (Smolensky, 1990), with spatial masks selecting roles and operation posteriors supplying fillers. Probes (Alain & Bengio, 2017; Belinkov, 2022) and control tasks (Hewitt & Liang, 2019) motivate testing what a readout recovers under a specified supervision protocol. We distinguish known factorial structure from assignment learned using flat cell labels, then score unseen bindings with injective alignment. This framing complements identifiability analyses by focusing on an empirical question of compositional readout under controlled supervision. Appendix J discusses predictive models, disentanglement, and causal representation learning in further detail.

3 Support×Operation Factorization

Frozen encoder and token maps. Let Φ be a frozen vision transformer that maps an image to a grid of patch tokens $\mathbf{z} \in \mathbb{R}^{N\times d}$. We use DINOv3-ViT-L (Siméoni et al., 2025) for the main experiments and SigLIP2-L/16 (Tschannen et al., 2025) for the cross-encoder replication. Both produce 1,024-dimensional tokens; their respective token grids contain 196 and 256 patches. The encoder is never trained or fine-tuned; all learning occurs in the readout.

The innovation field. Given a source image x_{src} and an intervened image x_{int} that differs by a single visual edit (e.g., a shifted floor hue), the *innovation field* is

$$\Delta\mathbf{z} = \Phi(x_{\mathrm{int}}) - \Phi(x_{\mathrm{src}}) \ \in \ \mathbb{R}^{N\times d}. \tag{1}$$

It records how the frozen token map changes under one intervention. The readout uses this field to characterize the change and source-image tokens to route patches to support slots (Eq. 4).

Supports and operations. We decompose the innovation field along two axes:

- A **support** $s \in \mathcal{S}$ is a ground-truth spatial region—a subset of patches where $\Delta\mathbf{z}$ typically concentrates: *where* the change occurs (floor, wall, or object).
- An **operation** $o \in \mathcal{O}$ is a transform class: *what* the change is (hue rotation, color inversion, or stripe overlay).

Predictive factors as support×operation cells. A predictive factor is a *cell* (s, o) in the factorial grid $\mathcal{S} \times \mathcal{O}$. With $|\mathcal{S}|$=3 supports and $|\mathcal{O}|$=3 operations, the grid contains nine cells. This definition

departs from analyses that equate a factor with a single global direction in latent space: it binds a spatial locus to a transform identity. The evaluation tests cell routing, while mask overlap with the ground-truth region is reported as a separate localization diagnostic.

The cell view makes spatial binding explicit: each support has a mask and each operation a slot. A global attribute direction $\mathbf{v}_k \in \mathbb{R}^d$, shared across patches, does not by itself specify this support–operation pairing.

4 Compositional Metric with Injective Binding

The evaluation asks whether a readout routes an unseen support×operation binding while distinguishing genuinely separate operation slots from degenerate slot sharing.

Leave-one-cell-out protocol. For each cell (s^*, o^*) in an $|\mathcal{S}| \times |\mathcal{O}|$ grid, we train on the other $|\mathcal{S}||\mathcal{O}| - 1$ cells. The support s^* remains paired with other operations and o^* remains paired with other supports; only their binding is held out. Within each observed cell, a seeded permutation allocates 64% of pairs to optimization, 16% to restart validation, and leaves 20% unused. Every pair from the held-out cell is test-only.

Joint prediction and axis alignment. For a batch of test pairs, the readout produces $E \in \mathbb{R}^{B\times|\mathcal{S}|\times|\mathcal{O}|}$ and predicts

$$(\hat{s}, \hat{o}) = \arg\max_{s,o} E_{b,s,o}. \tag{2}$$

The internal energy axes may be permuted, especially when the cell-to-grid assignment is learned. Using only optimization pairs, we therefore compute mean support- and operation-marginal energy matrices C^{sup} and C^{op} and bind each ground-truth axis value to an energy slot. This alignment is used for evaluation, never for training or restart selection.

Operation laundering. A non-injective alignment maps every axis value independently to its highest-energy slot. Two operations can then share a slot and both receive credit. We call this failure *operation laundering*. Injective alignment prevents it by requiring each ground-truth axis value to claim a distinct slot.

Hungarian injective assignment. The injective cell accuracy (INJ) constrains each axis map to a bijection:

$$\sigma^*_{\text{ax}} = \arg\max_{\sigma} \sum_{v} C^{\text{ax}}_{v,\sigma(v)}, \qquad \text{ax} \in \{\text{sup}, \text{op}\}. \tag{3}$$

The Hungarian algorithm (Kuhn, 1955) solves this assignment. A held-out example is correct when its joint argmax is $(\sigma^*_{\text{sup}}(s^*), \sigma^*_{\text{op}}(o^*))$. INJ is averaged first over all nine held-out cells within a seed and then across seeds.

Appendix H gives the complete evaluation algorithm.

Restart selection. Every restart is selected by flat observed-cell validation accuracy under the assignment fitted in that restart. Factorial assignment is fixed; learned assignment is optimized jointly with the readout. Validation does not use Hungarian alignment, held-out pairs, or held-out labels. The supervised MLP, attention, and oracle-operation baselines use native joint validation accuracy.

Diagnostics. One fit is a (seed, held-out cell) combination, giving 90 fits in a 10-seed 3×3 experiment. The *collapse rate* is the fraction of fits whose held-out-cell accuracy is below a predeclared threshold of 0.5; we also report thresholds 0.4 and 0.6 in the supplementary material. Assignment recovery compares a learned Q with the true factorial grid up to global permutations of the two axes. It is a diagnostic of recovered grid structure, not an input to training or scoring.

5 METHOD: FACTORED ENERGY AND GRID ASSIGNMENT

5.1 THE SO-OPF READOUT

A readout must distinguish where change is concentrated from which operation produced it. A dense readout that pools per-channel squared innovation couples these two roles, letting different operations share response channels. SO-OPF factors them apart, mapping an innovation field $\Delta\mathbf{z} \in \mathbb{R}^{N\times d}$ to a support×operation energy grid $E \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{O}|}$. For each held-out-cell fit, we standardize source tokens and innovations channelwise using statistics computed over all patches from optimization pairs; validation and test pairs reuse those statistics. We denote the standardized quantities by $\widetilde{\mathbf{z}}$ and $\widetilde{\Delta\mathbf{z}}$. A learned router first assigns patches to support slots:

$$M_{s,n} = \mathrm{softmax}_s(\langle W_z\widetilde{\mathbf{z}}_{\mathrm{src},n} + \mathbf{b}_z + \mathbf{p}_n, \mathbf{q}_s\rangle)\,, \tag{4}$$

where W_z, positional embeddings $\mathbf{p}_n$, and support queries $\mathbf{q}_s$ are learned. The router receives no pixel mask.

We additionally evaluate two router controls. The *position-only* router removes the source-token term from Eq. 4, while the *source-only* router removes the explicit positional embedding. The latter still inherits positional information from the frozen encoder and should therefore be interpreted as removing only the router-specific position term.

The readout then separates the amount of change from its identity:

$$E_{s,o} = A_s\,\pi_{s,o}, \tag{5}$$

with

$$A_s = \sum_n M_{s,n}\,\mathrm{sal}_n, \tag{6}$$

$$\pi_{s,o} = \mathrm{softmax}_o(f_\theta(\mathbf{h}_s)/\tau)\,, \tag{7}$$

$$\mathbf{h}_s = \frac{\sum_n M_{s,n}\mathrm{sal}_n\,\mathrm{LN}(\widetilde{\Delta\mathbf{z}}_n)}{\sum_n M_{s,n}\mathrm{sal}_n + 10^{-6}}, \tag{8}$$

where $\mathrm{sal}_n = \|\widetilde{\Delta\mathbf{z}}_n\|_2^2$. Thus A_s retains support-level innovation mass, while $\pi_{s,o}$ competes over operation identities using scale-normalized features. f_θ is a one-hidden-layer MLP with 64 GELU units, and τ is annealed from 2.0 to 0.5.

5.2 ASSIGNMENT SEMANTICS

For a held-out cell, let $K = |\mathcal{S}||\mathcal{O}| - 1$ be the number of observed cells. Each training pair has a flat label $y \in \{0,\dots,K-1\}$. A row-stochastic assignment $Q \in [0,1]^{K\times|\mathcal{S}||\mathcal{O}|}$ maps those labels to energy-grid locations. With normalized energy $\mathbf{sh} = E/(\sum_{s,o} E_{s,o} + 10^{-8})$, the observed-cell loss is

$$\mathcal{L}_{\mathrm{cell}} = \mathrm{CE}\big(\kappa\,\mathbf{sh}Q^\top, y\big)\,. \tag{9}$$

The positive logit scale κ controls the sharpness of the bounded energy shares. We keep it fixed across all assignment modes and readout comparisons; its value is reported in Appendix A.

We distinguish three assignment semantics.

- **Factorial Q.** Row y is a one-hot vector at the true grid location (s_y, o_y). This arm receives the known support–operation taxonomy and serves as the known-structure reference.
- **Learned Q.** Each row is $\mathrm{softmax}(\mathbf{a}_y)$ and is optimized jointly with the readout. Grid dimensions are known, but the core objective does not receive separate support or operation labels.
- **Compressed identity.** Observed label y maps to grid column y, leaving the final column unused. Because the held-out factorial cell changes across folds, this convention generally preserves neither grid axis. It tests the literal label-order interpretation, not a reindexing of factorial Q.

Held-out cells retain their true factorial coordinates under factorial Q; only compressed identity reserves the final slot. The metric's Hungarian alignment subsequently evaluates energy-slot semantics from optimization pairs and is applied only after training and restart selection.

5.3 Core and Axis-Aware Objectives

The core protocol uses

$$\mathcal{L}_{\text{core}} = \mathcal{L}_{\text{cell}} + \lambda_{\text{mask}}\mathcal{L}_{\text{mask}}, \tag{10}$$

where $\mathcal{L}_{\text{mask}}$ is the negative entropy of mean support usage and discourages empty support slots, and λ_{mask} controls its strength. This objective uses flat cell labels without native axis labels. We keep λ_{mask} fixed across all core comparisons; its value and the complete optimization settings are reported in Appendix A.

Some readout comparisons use a separate axis-aware enhanced protocol:

$$\begin{aligned}\mathcal{L}_{\text{enh}} = \mathcal{L}_{\text{cell}} + \lambda_{\text{loc}}\mathcal{L}_{\text{loc}} + \lambda_{\text{bal}}\mathcal{L}_{\text{bal}} \\ + \lambda_{\text{axis}}\mathcal{L}_{\text{axis}} + \lambda_{\text{mask}}\mathcal{L}_{\text{mask}} + \lambda_{\text{opinj}}\mathcal{L}_{\text{opinj}}.\end{aligned} \tag{11}$$

$\mathcal{L}_{\text{loc}}$ penalizes energy outside the cell assigned to a flat label and enforces a minimum assigned mass controlled by ρ. $\mathcal{L}_{\text{bal}}$ and $\mathcal{L}_{\text{axis}}$ discourage unused assignment columns and axes. $\mathcal{L}_{\text{opinj}}$ forms mean operation-slot profiles from *native operation labels* and penalizes their off-diagonal cosine similarity. Results using this loss compare readout design under axis-aware supervision, whereas the core protocol isolates assignment recovery from flat labels. All enhanced-protocol coefficients and ρ are fixed across comparisons and reported in Appendix A.

5.4 Baseline Readouts

Dense-Linear. The original axis-aware comparison uses a nonnegative linear readout that pools squared innovation channels independently for every operation:

$$E_{s,o}^{\text{dense}} = \sum_n M_{s,n} \sum_c \text{softplus}(W_{o,c})(\widetilde{\Delta \mathbf{z}}_{n,c})^2. \tag{12}$$

We refer to this historical baseline as *Dense-Linear*; it is compared with SO-OPF under matched axis-aware supervision.

Matched-core baselines. *Dense-MLP* applies a shared two-layer MLP to squared standardized patch innovations before support-weighted pooling. *Joint-MLP* concatenates all support summaries and relative support saliences and directly predicts the complete energy grid. Both use the same router, learned assignment, core objective, fixed optimization budget, and validation-only restart selection as SO-OPF; Dense-MLP is exactly and Joint-MLP is approximately parameter matched. Neither receives native axis labels. Appendix C.1 gives the complete architectures and parameter counts.

6 Experiments

6.1 Evaluation Settings and Protocol

The three evaluation settings allow us to examine controlled factor composition, sensitivity to rendering conditions, and behavior on natural images.

Shapes3D-Extended. We extend Shapes3D (Burgess & Kim, 2018) with three supports (floor, wall, object) and three operations (hue, invert, pattern), giving nine cells with 350 source–intervention pairs per cell. Hue selects a different native hue uniformly; inversion uses $255 - x$ within the support; pattern adds randomly oriented stripes with frequency in $[0.15, 0.45)$ and integer amplitude 40–89. We refer to the resulting benchmark as Shapes3D-Extended (S3D).

MuJoCo (alternative rendering and edit setting). To test sensitivity to a second controlled visual environment, we build an evaluation setting using MuJoCo (Todorov et al., 2012) with box geometry and separate camera and lighting settings, with 350 pairs per cell. Hue rotates the target material by a value sampled uniformly from $[0.25, 0.75)$ turns, and inversion changes the material before rerendering. Pattern remains a masked pixel-space stripe overlay with frequency $[0.08, 0.22)$ and amplitude 45–94.

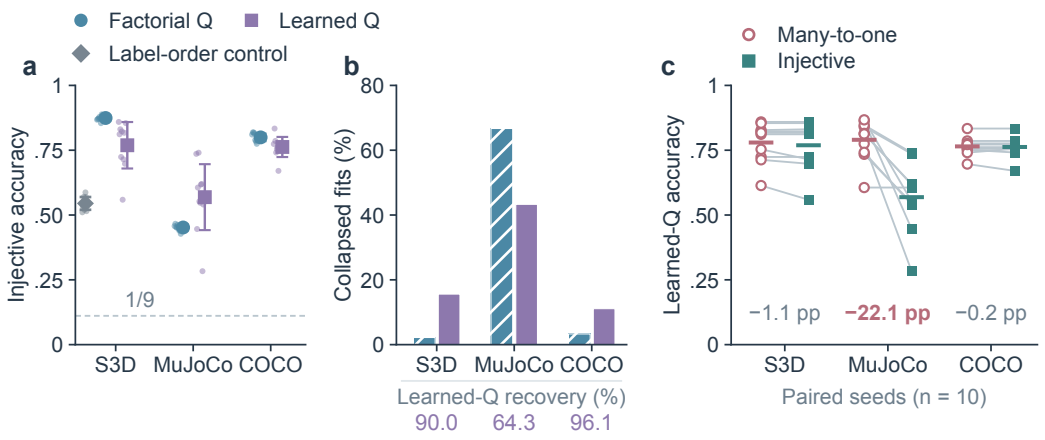

Figure 2: **DINOv3 assignment comparison. a**, Ten seed means over nine held-out cells, with mean and population SD; dashed line: chance. The label-order control (compressed identity) applies only to S3D. **b**, Collapse rates over 90 seed–cell fits (accuracy < 0.5); lower row: learned-Q grid recovery. **c**, Paired scores; short rules mark means, and annotations give injective minus many-to-one accuracy in percentage points (pp). MuJoCo loses > 5 pp in 9/10 seeds. All arms use the core protocol: factorial Q receives the axis taxonomy; learned Q uses flat cell labels.

Image-disjoint COCO. We edit person, vehicle, and background regions in COCO (Lin et al., 2014). Each cell contains 300 pairs, and all 2,700 source images are globally unique across cells; consequently no source used by an observed cell appears in the held-out cell. Hue is sampled uniformly from $[0.15, 0.85)$ turns, with the same inversion and stripe families as Shapes3D-Extended.

Features and optimization. We evaluate two frozen vision encoders using their bundled image processors and patch tokens (Siméoni et al., 2025; Tschannen et al., 2025). Restart selection and the observed-cell split follow the metric protocol above. Exact checkpoints, token dimensions, optimization settings, seeds, and hardware are reported in Appendix G.

Reporting. The assignment audit uses one fixed data construction and ten training seeds; we aggregate the nine held-out cells within each seed and report the across-seed mean and population SD. The matched-core comparison instead uses three data-construction seeds and three training/split seeds per construction. We report these sources of variation separately rather than treating training runs or held-out cells as independent data replicates.

Comparators. We compare a bilinear structured readout using factorial assignment, two-layer MLP and attention probes on mean-pooled or token-level $\Delta\mathbf{z}$, and an oracle-region operation classifier. The supervised probes receive native axis labels. Randomly initialized frozen ViT and raw-pixel readouts are additionally evaluated on COCO. The new matched-core comparison uses Dense-MLP and Joint-MLP under the same flat-label objective and fixed training budget as SO-OPF.

A support-localization diagnostic is reported in Appendix I (Figure 3); it is distinct from factorial assignment recovery.

6.2 Known Structure and Learned Structure

Factorial assignment establishes what the SO-OPF readout can do when the grid taxonomy is supplied. It composes held-out cells on Shapes3D-Extended and COCO. Learned assignment evaluates the more weakly supervised setting in which only flat cell labels are available. On Shapes3D-Extended, its raw and injective scores are 0.780 and 0.769; the paired factorial-minus-learned raw difference is 0.095 within the fixed data construction. On COCO, its raw and injective scores are 0.765 and 0.762. Recovery rates of 90.0% and 96.1% show that most fits discover an equivalent grid, but the score gaps and across-seed variation show that learned assignment remains distinguishable from known structure.

Table 1: **Assignment semantics under the core protocol.** INJ is injective cell accuracy; Min is the minimum held-out-cell mean; Coll. is the fraction of 90 seed–cell fits below 0.5; Rec. is factorial assignment recovery up to axis permutations. Values after ± are across-seed SD ($n = 10$). Figure 2 shows the DINOv3 rows. Factorial Q receives the axis taxonomy; learned Q uses flat cell labels only.

Assignment	INJ	Min	Coll.↓	Rec.↑
S3D, DINOv3				
factorial	.874±.009	.613	.022	n/a
learned	.769±.090	.560	.156	.900
compressed identity	.545±.025	.016	.444	n/a
MuJoCo, DINOv3				
factorial	.452±.011	.090	.667	n/a
learned	.569±.127	.124	.433	.643
MuJoCo, SigLIP2				
factorial	.476±.008	.045	.656	n/a
learned	.484±.167	.003	.511	.676
COCO, DINOv3				
factorial	.799±.015	.524	.033	n/a
learned	.762±.039	.533	.111	.961
COCO, SigLIP2				
factorial	.826±.012	.563	.011	n/a
learned	.751±.074	.448	.078	.965

The same COCO pattern replicates with SigLIP2. Factorial and learned assignments reach 0.826 and 0.751 INJ; their paired difference is 0.075 within the fixed data construction. Learned assignment recovers an equivalent grid in 96.5% of fits. Thus changing the frozen encoder preserves both the strong readout result and the measurable cost of estimating its assignment from flat labels.

Injective scoring exposes operation laundering. Without injective binding, a many-to-one alignment can credit two operations through the same slot and inflate accuracy. This effect is most visible on MuJoCo: learned assignment scores 0.790 under many-to-one alignment but drops to 0.569 after injective scoring—a gap of 0.221 that is absent from the aggregate many-to-one score (Figure 2c). The paired seed-level view shows this is not an averaging artifact: 9 of 10 MuJoCo seeds lose more than 0.05 accuracy under injective scoring, against 1 of 10 on Shapes3D-Extended and 0 of 10 on COCO. SigLIP2 exhibits the same pattern: its learned score falls from 0.629 to 0.484. These gaps confirm that many-to-one accuracy can mask substantial slot reuse.

Label-order control. Compressed identity tests whether label order alone supplies a grid. Mapping the i-th observed flat label to the i-th slot scores 0.545 and collapses in 44.4% of fits. Because a different cell is held out in every fold, the compressed mapping assigns different labels to the same row or column across folds; this inconsistency means that label order generally violates the grid's axis structure rather than simply reindexing it.

Renderer sensitivity. MuJoCo provides a more challenging test of generalization across intervention settings. With DINOv3, factorial assignment reaches 0.452; learned assignment rises to 0.790 before injective alignment but falls to 0.569 after it, with 43.3% collapse. The second encoder shows a similar pattern: factorial and learned INJ are 0.476 and 0.484, and the learned score drops from 0.629 to 0.484 under injective alignment. Its 51.1% collapse and minimum cell mean of 0.003 indicate strong fold dependence. Per-cell breakdowns reveal that pattern cells remain near-perfect (> 0.99) while hue and inversion cells account for most of the difficulty—consistent with material-space operations producing less discriminable token-level signatures than pixel-space overlays. This pattern is stable across collapse thresholds 0.4–0.6 (Appendix F). The cross-encoder agreement identifies the intervention setting as an important condition for reliable compositional readout.

Table 2: **Matched-core readout comparison.** INJ means over three data constructions with three training/split seeds each. All methods use flat labels and a shared fixed budget; Dense-MLP is exactly and Joint-MLP approximately parameter matched.

Setting	SO-OPF	Dense-MLP	Joint-MLP
DINOv3 / COCO	.754	.275	.120
SigLIP2 / COCO	.778	.269	.127
DINOv3 / MuJoCo	.554	.286	.094
SigLIP2 / MuJoCo	.492	.192	.077

6.3 Baselines and Readout Comparison

For each encoder–dataset pair and each of its three data constructions, SO-OPF has higher mean INJ than both controls after averaging the three training/split seeds and held-out cells. Appendix C reports per-construction differences, fit diagnostics, and conditional source-bootstrap intervals. The component effects are not uniform across datasets (Appendix E.2).

Position and edit controls. Fixed-layout controls reveal different routing regimes: position-only routing is strong on Shapes3D-Extended, whereas source features are more informative on COCO. On unseen layouts, source-based routing outperforms position-only routing, with the source-only variant giving the strongest result. Edit-family controls remain stable across new pixel-space parameters and waveforms, while transfer to material-space edits is more challenging (Appendix D).

7 Limitations

Scope of recovery. The core learned-Q arm studies weakly supervised assignment recovery: grid dimensions, flat cell labels, and the intervention construction are known, but separate axis labels are not. Factorial assignment receives the full support–operation taxonomy, and the enhanced op-inj objective receives native operation labels. Learned assignment retains much of the factorial-Q score on Shapes3D-Extended and COCO, including the SigLIP2 replication, but does not address unknown grid dimensions or fully unsupervised factor discovery.

Intervention and evaluation scope. Position-only routing performs strongly in some fixed-layout settings, whereas randomized-layout controls cover static synthetic rearrangements. Extending the framework to video motion, open-ended supports, and semantic operations remains future work. Statistical and optimization details are provided in Appendix C.

8 Conclusion

Readout capacity and assignment recovery are distinct requirements for compositional readout. SO-OPF reduces operation laundering relative to the tested Dense-MLP and Joint-MLP readouts under the same flat-label objective and fixed training budget, and routes held-out cells when the factorial grid is known. Injective scoring exposes slot reuse that many-to-one alignment conceals. Position, layout, and edit controls further distinguish position-driven routing from source-based generalization and characterize transfer across intervention realizations. Together, the experiments establish a controlled framework for reporting readout capacity, assignment recovery, and alignment separately.

Reproducibility Statement

Sections 4, 5, and 6.1 specify the evaluation splits, assignment modes, objectives, frozen encoders, and optimization settings. Appendices A and G provide additional protocol and reproducibility details, and Appendix F reports sensitivity to the collapse threshold and operation-specific failures.

AI USE STATEMENT

AI assistants were used to help refine the research questions and experimental design, develop and review experiment code, interpret results, and draft and revise the manuscript. They also assisted with the preparation and checking of this submission's LaTeX source. The authors reviewed all AI-assisted work and take responsibility for the final content of the submission.

REFERENCES

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. In *International Conference on Learning Representations (Workshop)*, 2017.

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *Conference on Computer Vision and Pattern Recognition*, pp. 15619–15629, 2023.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mido Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. *arXiv preprint arXiv:2404.08471*, 2024.

David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6541–6549, 2017.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022.

Chris Burgess and Hyunjik Kim. 3D shapes dataset. `https://github.com/deepmind/3d-shapes`, 2018.

Christopher P. Burgess, Loic Matthey, Nicholas Watters, Rishabh Kabra, Irina Higgins, Matt Botvinick, and Alexander Lerchner. MONet: Unsupervised scene decomposition and representation. *arXiv preprint arXiv:1901.11390*, 2019.

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

Klaus Greff, Raphaël Lopez Kaufman, Rishabh Kabra, Nick Watters, Christopher Burgess, Daniel Zoran, Loic Matthey, Matthew Botvinick, and Alexander Lerchner. Multi-object representation learning with iterative variational inference. In *International Conference on Machine Learning*, pp. 2424–2433, 2019.

John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In *Conference on Empirical Methods in Natural Language Processing*, 2019.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. β-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017.

Irina Higgins, David Amos, David Pfau, Sébastien Racanière, Loic Matthey, Danilo Rezende, and Alexander Lerchner. Towards a definition of disentangled representations. *arXiv preprint arXiv:1812.02230*, 2018.

Aapo Hyvärinen, Hiroaki Sasaki, and Richard Turner. Nonlinear ICA using auxiliary variables and generalized contrastive learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 859–868, 2019.

Daniel Keysers, Nathanael Schärli, Nathan Scales, Hylke Buisman, Daniel Furrer, Sergii Kashubin, Nikola Momchev, Danila Sinopalnikov, Lukasz Stafiniak, Tibor Tihon, Dmitry Tsarkov, Xiao Wang, Marc van Zee, and Olivier Bousquet. Measuring compositional generalization: A comprehensive method on realistic data. In *International Conference on Learning Representations*, 2020.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV). In *International Conference on Machine Learning*, pp. 2668–2677, 2018.

Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In *International Conference on Machine Learning*, pp. 2649–2658, 2018.

Thomas Kipf, Gamaleldin Elsayed, Aravindh Mahendran, Austin Stone, Sara Sabour, Georg Heigold, Rico Jonschkowski, Alexey Dosovitskiy, and Klaus Greff. Conditional object-centric learning from video. In *International Conference on Learning Representations*, 2022.

Yilun Kuang, Yash Dagade, Tim G. J. Rudner, Randall Balestriero, and Yann LeCun. Rectified LpJEPA: Joint-embedding predictive architectures with sparse and maximum-entropy representations. *arXiv preprint arXiv:2602.01456*, 2026.

Harold W Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955.

Brenden Lake and Marco Baroni. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *International Conference on Machine Learning*, pp. 2873–2882, 2018.

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

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision*, pp. 740–755, 2014.

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Rätsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *International Conference on Machine Learning*, pp. 4114–4124, 2019.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. In *Advances in Neural Information Processing Systems*, pp. 11525–11538, 2020.

Ishan Misra, Abhinav Gupta, and Martial Hebert. From red wine to red tomato: Composition with context. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1792–1801, 2017.

Milton Llera Montero, Casimir JH Ludwig, Rui Ponte Costa, Gaurav Malhotra, and Jeffrey Bowers. The role of disentanglement in generalisation. In *International Conference on Learning Representations*, 2021.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.

Lukas Schott, Julius von Kügelgen, Frederik Träuble, Peter Gehler, Chris Russell, Matthias Bethge, Bernhard Schölkopf, Francesco Locatello, and Wieland Brendel. Visual representation learning does not generalize strongly within the same domain. In *International Conference on Learning Representations*, 2022.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, et al. DINOv3. *arXiv preprint arXiv:2508.10104*, 2025.

Gautam Singh, Yi-Fu Wu, and Sungjin Ahn. Simple unsupervised object-centric learning for complex and naturalistic videos. In *Advances in Neural Information Processing Systems*, 2022.

Paul Smolensky. Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence*, 46(1–2):159–216, 1990.

Lucas Thil, Jesse Read, Rim Kaddah, and Guillaume Doquet. Subspace-decomposed JEPAs: Disentangling progression and content in latent world models. *arXiv preprint arXiv:2605.31111*, 2026.

Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033, 2012.

Frederik Träuble, Elliot Creager, Niki Kilbertus, Francesco Locatello, Andrea Dittadi, Anirudh Goyal, Bernhard Schölkopf, and Stefan Bauer. On disentangled representations learned from correlated data. In *International Conference on Machine Learning*, pp. 10401–10412, 2021.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. SigLIP 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

Julius von Kügelgen, Yash Sharma, Luigi Gresele, Wieland Brendel, Bernhard Schölkopf, Michel Besserve, and Francesco Locatello. Self-supervised learning with data augmentations provably isolates content from style. In *Advances in Neural Information Processing Systems*, 2021.

Thaddäus Wiedemer, Prasanna Mayilvahanan, Matthias Bethge, and Wieland Brendel. Compositional generalization from first principles. In *Advances in Neural Information Processing Systems*, 2023.

A Additional Protocol and Results

Protocol details. Compressed identity does not reindex the held-out cell to the ninth factorial coordinate. Within each observed cell a seeded permutation first allocates 80% of pairs to a development pool, from which 20% is withheld for validation—yielding 224/56/70 optimization/validation/unused pairs per 350-pair cell (192/48/60 per 300-pair COCO cell). For every mode, restart selection uses flat observed-cell validation accuracy; Hungarian alignment and held-out pairs are unavailable during selection.

Core and enhanced supervision. All protocols use the fixed cell-logit scale $\kappa = 10$. The core objective uses unit-weight cell cross-entropy and support-use entropy; localization, assignment-balance, axis-balance, and operation-injectivity weights are zero. The enhanced objective adds localization (weight 1), assignment balance (0.1), axis balance (0.5), and operation injectivity (1), with localization margin $\rho = 0.6$. Operation injectivity uses native operation labels, making the enhanced protocol axis-aware.

B Cross-Encoder Consolidated Summary

Table 3 consolidates all encoder×dataset ×assignment results to facilitate pattern comparison.

Table 3: **Cross-encoder summary.** INJ is injective cell accuracy; Coll. is collapse rate at threshold 0.5; Rec. is factorial assignment recovery. All values are means over $n = 10$ seeds.

Dataset	Encoder	Assign.	INJ	Coll.	Rec.
S3D	DINOv3	factorial	.874	.022	–
S3D	DINOv3	learned	.769	.156	.900
S3D	DINOv3	compressed	.545	.444	–
COCO	DINOv3	factorial	.799	.033	–
COCO	DINOv3	learned	.762	.111	.961
COCO	SigLIP2	factorial	.826	.011	–
COCO	SigLIP2	learned	.751	.078	.965
MuJoCo	DINOv3	factorial	.452	.667	–
MuJoCo	DINOv3	learned	.569	.433	.643
MuJoCo	SigLIP2	factorial	.476	.656	–
MuJoCo	SigLIP2	learned	.484	.511	.676

On Shapes3D-Extended and COCO, learned assignment approaches factorial performance with low collapse and high recovery. On MuJoCo, both encoders show high collapse and low INJ regardless of assignment mode, showing that the original intervention setting is challenging across encoder families. Because that setting also changes how edits are implemented, it is not a pure renderer-transfer comparison.

C Matched-Core Baselines and Statistical Replication

C.1 Preprocessing and baseline definitions

For each held-out-cell fit, channelwise means and sample standard deviations are computed over all patches from optimization pairs, separately for source tokens and innovations. A constant 10^{-6} is added to each standard deviation, and the resulting statistics are reused for validation and test pairs. All readouts use the router in Eq. 4; the router parameters are trained separately for each method rather than shared across fitted models. Layer normalization in the support summaries is tokenwise over feature channels, uses 10^{-5} numerical stabilization, and has no learned affine parameters.

Dense-MLP applies a shared MLP to each squared standardized patch innovation:

$$\mathbf{x}_n = \widetilde{\Delta\mathbf{z}}_n \odot \widetilde{\Delta\mathbf{z}}_n, \quad \mathbf{g}_n = \mathrm{GELU}(W_1\mathbf{x}_n + \mathbf{b}_1), \quad \mathbf{e}_n = \mathrm{softplus}(W_2\mathbf{g}_n + \mathbf{b}_2), \tag{13}$$

where the dimensions are $1024 ightarrow 64 \to 3$. Its energy is $E_{s,o} = \sum_n M_{s,n} e_{n,o}$. It has no operation softmax or explicit support-salience product.

Joint-MLP first forms the same support summaries as SO-OPF and relative support saliences

$$\bar{A}_s = A_s / (\sum_{s'} A_{s'} + 10^{-8}), \qquad \mathbf{v} = [\mathbf{h}_1^\top, \ldots, \mathbf{h}_S^\top, \bar{A}_1, \ldots, \bar{A}_S]^\top. \tag{14}$$

For $d = 1024$ and $S = 3$, $\mathbf{v}$ has dimension 3075. A $3075 \to 21 \to 9$ MLP with GELU hidden units and elementwise Softplus outputs predicts the full energy grid in support-major order. It has no operation softmax or explicit salience–posterior product.

The matched-core comparison uses the same learned assignment, flat cell labels, core objective, 2,000 optimization steps, five restarts, and validation-only restart selection for every readout. The dominant router architecture is common, Dense-MLP is exactly parameter matched, and Joint-MLP is approximately parameter matched:

Table 4: **Trainable readout parameters in the matched-core comparison.** The learned assignment adds 72 parameters to every row.

Encoder	Readout	Router	Head	Total
DINOv3	SO-OPF	1,253,376	65,795	1,319,171
DINOv3	Dense-MLP	1,253,376	65,795	1,319,171
DINOv3	Joint-MLP	1,253,376	64,794	1,318,170
SigLIP2	SO-OPF	1,314,816	65,795	1,380,611
SigLIP2	Dense-MLP	1,314,816	65,795	1,380,611
SigLIP2	Joint-MLP	1,314,816	64,794	1,379,610

C.2 Observed-cell fit diagnostics

Table 5 reports terminal observed-cell training and validation accuracy for the selected restart, together with held-out INJ. The values average three data constructions, three training/split seeds per construction, and nine held-out cells. Observed-cell classification has eight classes and is distinct from nine-cell held-out evaluation.

Table 5: **Terminal fit diagnostics for the matched-core comparison.** All entries are percentages.

Setting	Readout	Train	Validation	Held-out INJ
DINOv3 / COCO	SO-OPF	99.50	84.69	75.40
	Dense-MLP	83.68	57.28	27.52
	Joint-MLP	86.15	81.17	12.01
DINOv3 / MuJoCo	SO-OPF	93.32	81.50	55.35
	Dense-MLP	85.57	72.63	28.55
	Joint-MLP	63.76	60.48	9.42
SigLIP2 / COCO	SO-OPF	99.55	87.59	77.79
	Dense-MLP	85.80	62.14	26.93
	Joint-MLP	84.19	79.40	12.73
SigLIP2 / MuJoCo	SO-OPF	91.91	78.22	49.15
	Dense-MLP	77.16	72.24	19.22
	Joint-MLP	51.24	50.04	7.71

The lower Joint-MLP training accuracy on learned-assignment MuJoCo shows that optimization and assignment learning contribute to its weak held-out result. A separate DINOv3 known-assignment sanity check shows that Joint-MLP can fit the observed task while still failing to compose:

These diagnostics use saved terminal energies and selected-restart validation scores. Stepwise learning curves were not retained and no learning-rate sweep was performed, so the comparison is reported as a shared fixed-budget evaluation rather than evidence that every baseline is optimally tuned.

Table 6: **Joint-MLP known-assignment sanity check on the cached fixed data.** Values are percentages over three training seeds and nine held-out cells.

Dataset	Train	Validation	Held-out INJ
Shapes3D-Extended	100.00	98.93	0.00
COCO	100.00	95.56	0.00
MuJoCo	100.00	85.76	0.01

For each encoder–dataset pair, three fresh data constructions are trained with three training/split seeds. Table 7 reports the SO-OPF minus dense difference for each construction. The intervals resample paired source-level predictions from the already trained models; they are conditional intervals and do not include retraining uncertainty.

Table 7: **Per-construction matched-core differences in percentage points.** Brackets give conditional 95% paired source-bootstrap intervals; SD is the sample SD of the three construction-level differences.

Setting	Data 0	Data 1	Data 2	SD
DINOv3 / COCO	47.32 [46.04,48.73]	49.36 [48.09,50.62]	46.95 [45.50,48.27]	1.30
SigLIP2 / COCO	56.27 [54.99,57.63]	48.62 [47.38,49.94]	47.72 [46.49,49.01]	4.70
DINOv3 / MuJoCo	25.89 [25.21,26.54]	27.44 [26.88,27.99]	27.07 [26.46,27.65]	0.81
SigLIP2 / MuJoCo	29.14 [28.58,29.69]	40.60 [40.05,41.19]	20.05 [19.58,20.56]	10.30

Table 8 gives the corresponding Joint-MLP comparison.

Table 8: **Per-construction SO-OPF minus Joint-MLP differences in percentage points.** Intervals and SD follow Table 7.

Setting	Data 0	Data 1	Data 2	SD
DINOv3 / COCO	67.00 [65.85,68.15]	60.54 [59.40,61.69]	62.62 [61.49,63.78]	3.30
SigLIP2 / COCO	64.65 [63.53,65.79]	66.26 [65.17,67.36]	64.27 [63.13,65.48]	1.05
DINOv3 / MuJoCo	40.22 [39.62,40.83]	45.66 [45.15,46.17]	51.90 [51.36,52.42]	5.85
SigLIP2 / MuJoCo	32.40 [31.99,32.85]	58.02 [57.56,58.47]	33.88 [33.56,34.22]	14.38

The direction is consistent across the tested constructions, while the variation for SigLIP2 on MuJoCo cautions against interpreting the narrow conditional intervals as dataset-level certainty. The three COCO constructions draw from a shared finite image pool, and three data seeds remain insufficient for a strong population-level significance claim.

For completeness, Table 9 retains the original baselines, which use different structural inputs and are not substitutes for the matched-core comparison.

D Position, Layout, and Edit Controls

Fixed-layout router controls. Table 10 uses the learned-assignment core protocol with three training/split seeds. Position-only removes source-token input from the router but leaves innovation features in the operation readout. Source-only removes the router-specific positional embedding, not positional information already present in the frozen encoder.

The Shapes3D-Extended and MuJoCo results show that fixed-layout accuracy alone cannot establish semantic support localization. These three-seed controls are distinct from the ten-seed assignment audit in Table 1 and should not be numerically merged with it.

Unseen layouts. We randomize sphere, box, and cylinder positions and use disjoint training and test permutation sets. Table 11 reports data construction 0; two additional independently generated test scenes give the same ordering but reuse the trained models. The support order is sphere, box, cylinder.

Table 9: **Original core-protocol baselines grouped by structural input.** Entries are INJ means over ten training seeds on the fixed data construction.

Method	S3D	MuJoCo	COCO
Known factorial assignment			
SO-OPF	.874	.452	.799
Bilinear, factorial Q	.742	.428	.601
Flat observed-cell labels			
SO-OPF, learned Q	.769	.569	.762
Native-axis supervised probes			
MLP probe	.056	.004	.049
Attention probe	.160	.024	.117
Oracle-region operation	.784	.387	.310
Alternative COCO readouts, factorial Q			
Random-init ViT	–	–	.573
Pixels	–	–	.503

Table 10: **DINOv3 fixed-layout router controls.** Values are mean INJ percentages with sample SD across three training/split seeds.

Dataset	Full SO-OPF	Position-only	Source-only
Shapes3D-Extended	77.85 ± 12.75	84.12 ± 6.70	75.48 ± 2.58
COCO	76.96 ± 0.51	42.27 ± 7.13	74.79 ± 3.35
MuJoCo	50.37 ± 15.66	50.32 ± 5.97	53.34 ± 10.47

Their respective sizes are radius 0.28; box half-lengths $(0.24, 0.24, 0.24)$; and cylinder radius 0.24 with half-length 0.28. The camera is fixed at $(0, 0, 4)$ and images are rendered at 224×224. For a slot permutation p,

$$x_s = 0.85(p_s - 1) + \epsilon_s^x, \quad \epsilon_s^x \sim U[-0.06, 0.06), \quad y_s \sim U[-0.18, 0.18), \quad z_s = 0. \tag{15}$$

The training permutations are $(0, 1, 2)$, $(1, 2, 0)$, and $(2, 0, 1)$; the unseen test permutations are $(0, 2, 1)$, $(2, 1, 0)$, and $(1, 0, 2)$. Each construction contains 2,700 pairs, exactly 300 per support–operation cell. A permutation is sampled uniformly for each scene rather than forced to equal counts. Layout randomness is separate from source appearance and edit randomness.

Table 11: **Randomized-layout training and unseen-layout testing.** Values are mean INJ percentages with sample SD over three training/split seeds.

Encoder	Full SO-OPF	Position-only	Source-only
DINOv3	57.67 ± 6.43	35.42 ± 5.41	67.09 ± 4.24
SigLIP2	83.67 ± 7.40	7.54 ± 10.85	87.74 ± 4.94

Source features improve substantially over position-only routing in this layout shift, but the explicit router position term is not necessary and can be detrimental. Fixed-layout training followed by unseen-layout testing is harder, reaching 47.77% with DINOv3 and 43.02% with SigLIP2.

Edit-family and edit-space controls. All three evaluation settings are rebuilt with the same pixel-space HSV rotation, inversion, and stripe implementation. Table 13 evaluates the trained readout on the original pixel family, unseen stripe parameters, and a checker waveform. Hue adds a shift sampled uniformly from $[0.2, 0.8)$ modulo one in HSV space, while inversion applies $255 - I$ inside the selected support. For normalized image coordinates, define

$$u = (x/W)\cos\theta + (y/H)\sin\theta, \qquad v = -(x/W)\sin\theta + (y/H)\cos\theta. \tag{16}$$

The stripe waveform is $\mathrm{sign}(\sin(2\pi f u+\phi))$; the checker waveform is $\mathrm{sign}(\sin(2\pi f u+\phi)\sin(2\pi f v+\phi))$. An integer amplitude is added equally to the three RGB channels and the result is clipped to the valid image range.

Table 12: **Pixel-space pattern distributions.** Angles are in radians and frequencies are cycles in normalized image coordinates.

Parameter	Train stripe	Unseen stripe	Checker
θ	$U[0,\pi/2)$	$U[\pi/2,\pi)$	$U[0,\pi/2)$
f	$U[6,12)$	$U[12,18)$	$U[6,12)$
ϕ	$U[0,2\pi)$	$U[0,2\pi)$	$U[0,2\pi)$
Amplitude	integers 40–89	integers 40–89	integers 40–89
Waveform	stripe	stripe	checker

The unseen-stripe control changes orientation and frequency, whereas the checker control keeps the training ranges and changes only the waveform. Source selection, layout, and edits use separate random streams. Each unified-pixel construction contains 300 pairs per cell; the native MuJoCo multi-construction comparison instead uses 350 pairs per cell.

Table 13: **Transfer within the pixel-space edit family.** Entries are mean INJ percentages over three training/split seeds.

Setting	Original	Unseen stripe	Checker
DINOv3 / S3D	57.91	57.88	52.62
DINOv3 / COCO	65.49	66.85	65.22
DINOv3 / MuJoCo	79.86	78.63	74.73
SigLIP2 / COCO	70.93	71.57	68.20
SigLIP2 / MuJoCo	90.02	90.04	90.02

The stability rules out dependence on one sampled stripe-parameter range, but hue, inversion, and pattern remain distinguishable low-level edit families. Keeping the MuJoCo renderer fixed while transferring a pixel-trained readout to material-space hue and inversion reduces INJ from 79.86% to 68.49% for DINOv3 and from 90.02% to 69.37% for SigLIP2. Thus the original cross-setting comparison conflates renderer and edit realization and is not evidence of pure renderer invariance.

E Scaling and Readout Controls

Larger operation set. Extending Shapes3D-Extended to a 3×5 grid (hue, invert, pattern, brightness, blur; 15 cells, chance $1/15$) gives INJ 0.870 ± 0.013, zero collapse, and distinct-support rate 1.0 under known factorial assignment. The minimum per-cell mean is 0.586.

Two-support simultaneous edits. When two observed supports are edited simultaneously, a top-2 readout over the energy tensor reaches joint accuracy 0.843 ± 0.006 over five seeds (uniform chance $1/36$), with at-least-one-correct accuracy 0.999 and support-pair agreement 0.833. This tests superposition at inference time rather than discovery of an unseen grid cell.

COCO readouts. On the rebuilt, globally image-disjoint COCO data, factorial SO-OPF reaches 0.799 ± 0.015, compared with 0.601 ± 0.017 for the bilinear readout, 0.573 ± 0.020 for a frozen randomly initialized ViT, and 0.503 ± 0.016 for pixels under injective alignment. Supervised MLP, attention, and oracle-operation probes reach 0.049, 0.117, and 0.310.

E.1 Matched readout comparison

Table 14 compares factorized, bilinear, and Dense-Linear readouts. The enhanced protocol uses native operation labels and is therefore an axis-aware readout comparison rather than a flat-label assignment-recovery experiment.

Table 14: **Shapes3D-Extended readout comparison over ten seeds.** Core and enhanced refer to Eqs. 10 and 11. Non-inj and INJ are many-to-one and injective scores; Coll. uses threshold 0.5.

Readout	Assignment	Protocol	Non-inj	INJ	Coll.
Factored	factorial	core	.874	.874	.022
Factored	learned	core	.780	.769	.156
Bilinear	factorial	core	.742	.742	.100
Factored	factorial	enhanced	.913	.913	.000
Factored	learned	enhanced	.841	.841	.000
Dense-Linear	learned	enhanced	.724	.653	.267

Under matched enhanced supervision, factorization removes the Dense-Linear readout's 0.071 laundering gap and improves learned-Q INJ from 0.653 to 0.841. With factorial assignment, the enhanced factored readout reaches 0.913; the paired factorial-minus-learned difference is 0.072 on the fixed data construction.

E.2 Component ablations

Table 15 reports DINOv3 learned-assignment core results over three training/split seeds on the cached fixed data. These runs diagnose components of the original parameterization and are not mixed with the fresh multi-construction matched-core estimates.

Table 15: **SO-OPF component ablations.** Entries are mean INJ; each column uses the same three training/split seeds.

Configuration	S3D	COCO	MuJoCo
SO-OPF	.779	.770	.504
No output support salience	.186	.275	.244
No salience-weighted pooling	.692	.685	.642
No operation softmax	.547	.600	.537
No scale normalization	.812	.786	.566
Uniform support router	.314	.211	.221
Source-only router	.755	.748	.533
Joint-MLP	.104	.092	.157

Output support salience and a learned router are important in these settings, but the remaining effects are not uniform: removing scale normalization improves the mean in all three columns, while the effects of salience-weighted pooling and operation competition vary by dataset. The evidence therefore supports the tested factorized readout as a whole rather than the universal necessity of every component. The no-salience variant also changes marginal energy structure, and the Joint-MLP changes the full parameterization, so neither is a pure single-mechanism intervention.

F Sensitivity and Failure Analysis

Collapse threshold robustness. At thresholds 0.4/0.5/0.6, collapse rates for factorial versus learned assignment are 1.1/2.2/3.3% versus 13.3/15.6/18.9% on Shapes3D-Extended, 0.0/3.3/18.9% versus 2.2/11.1/23.3% on COCO, and 66.7/66.7/66.7% versus 42.2/43.3/43.3% on rebuilt MuJoCo. The method ordering and the MuJoCo boundary do not depend on choosing exactly 0.5.

For the SigLIP2 replication, the corresponding factorial/learned rates at thresholds 0.4/0.5/0.6 are 0.0/1.1/7.8% versus 6.7/7.8/20.0% on COCO and 51.1/65.6/66.7% versus 51.1/51.1/52.2% on MuJoCo. COCO remains stable; MuJoCo remains failure-prone throughout the threshold range.

MuJoCo per-cell breakdown. Hue and inversion are applied by changing renderer materials before rerendering; pattern is a masked pixel-space overlay. Under DINOv3 factorial assignment, pattern

cells score 0.992–1.000, whereas hue cells score 0.090–0.275 and inversion cells are similarly low. SigLIP2 shows the same split: all three pattern cells score 1.000 while hue scores 0.236–0.439 and inversion 0.063–0.072. The boundary is therefore operation-specific, not uniform across cells. This pattern is consistent with material-space operations producing less discriminable token-level signatures than pixel-space overlays, though the present evidence does not isolate this as the sole cause.

Learned assignment on MuJoCo. Learned assignment improves aggregate DINOv3 INJ from 0.452 to 0.569 but recovers only 64.3% of factorial assignments and retains 43.3% collapse. Under SigLIP2, the learned score drops from 0.629 (many-to-one) to 0.484 (injective), with distinct-support rate 0.856 and 51.1% collapse. The injective drop confirms slot reuse rather than genuine compositional routing.

Statistical comparisons. The original assignment audit uses one fixed data construction and ten training seeds. Its factorial-minus-learned raw difference is 0.095 under the core protocol. Under the enhanced axis-aware protocol the difference is 0.072. These are conditional comparisons of training variation on fixed data, not estimates of population-level sampling uncertainty. Section C separately reports the new multi-construction comparison and conditional source-bootstrap intervals.

G Reproducibility Details

The original assignment audit uses training seeds 0–9, 2,000 optimization steps, five restarts, and test-blind validation selection. The matched-core experiment uses data-construction seeds 0–2 and three training/split seeds within each construction, with the same step and restart budget for every readout. Matched-core runs use full-batch Adam with learning rate 0.005, $(\beta_1, \beta_2) = (0.9, 0.999)$, $\epsilon = 10^{-8}$, and no weight decay, learning-rate schedule, early stopping, or minibatching. Learned-Q logits are initialized to zero; the final observed-cell validation accuracy selects among five restarts. SO-OPF alone uses operation-temperature annealing from 2 to 0.5; the direct-output baselines do not use this temperature. The DINOv3 and SigLIP2 checkpoints are `dinov3-vitl16-pretrain-lvd1689m` and `siglip2-large-patch16-256`. Their respective runs used NVIDIA H200 and H100 GPUs. The result artifacts retain assignment mode, validation selector, loss weights, seeds, restarts, and cache provenance. The software environment lock, renderer configuration, preprocessing code, and per-fit outputs will be released with the final publication.

H Evaluation Algorithm

Algorithm 1 Leave-one-cell-out injective evaluation

Input: Cell-partitioned data $\mathcal{D}$, seeds, restarts, assignment mode Q
Output: Per-cell and mean INJ

for seed k and held-out cell $h = (s^*, o^*)$ **do**
  Split each observed cell: 64% optimization, 16% validation, 20% unused.
  Reserve all pairs in h for test.
  for restart r **do**
    Fit the readout and, when applicable, learned Q.
    Score flat observed-cell accuracy on validation pairs.
  end for
  Select the highest-validation restart.
  No Hungarian alignment or test labels enter selection.
  From optimization pairs, compute $C^{\text{sup}}, C^{\text{op}}$.
  Compute $\sigma^*_{\text{sup}}, \sigma^*_{\text{op}}$ by Hungarian assignment.
  Evaluate the joint argmax on held-out-cell pairs.
end for
return Seed- and cell-aggregated INJ.

I SUPPORT LOCALIZATION DIAGNOSTIC

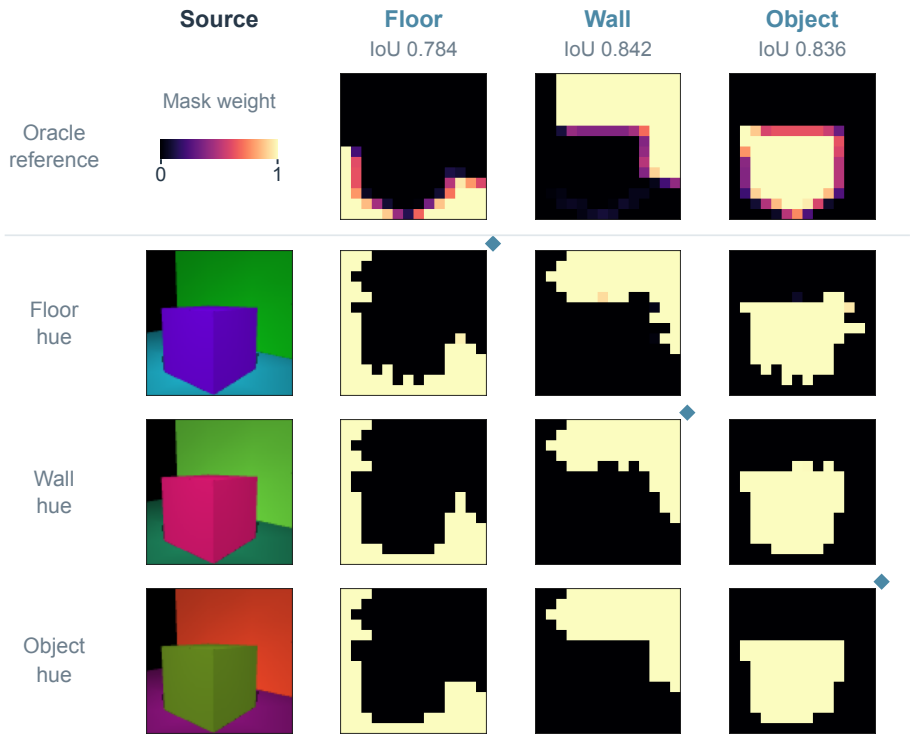

Figure 3: **MuJoCo support-router diagnostic.** Oracle reference patch weights (top) and learned softmax masks for three source images; diamonds mark the intervened support. After argmax assignment and Hungarian slot matching, the support IoUs above the columns summarize all 630 held-out examples, not the displayed images individually. Mean IoU is 0.821 and patch-assignment agreement is 0.903. This separate diagnostic trains on all nine cells with a stratified 80/20 split and op-inj using native operation labels. It evaluates support localization, not leave-one-cell-out composition or learned-Q factorial-assignment recovery.

J EXTENDED RELATED WORK

Disentangled representation learning. β-VAE (Higgins et al., 2017), FactorVAE (Kim & Mnih, 2018), and their descendants encourage factors to align with statistically independent latent coordinates. Locatello et al. (2019) show that unsupervised disentanglement is underdetermined without inductive bias; subsequent work reveals that even well-disentangled representations fail compositional generalization (Montero et al., 2021; Schott et al., 2022), and that factor correlations in the training data further entangle what is learned (Träuble et al., 2021). Higgins et al. (2018) ground disentanglement in symmetry transformations, closer to our operation axis. One source of the failure may be the global-direction definition itself, which provides no explicit binding between a spatial region and a transform class.

JEPA and predictive world models. Joint-Embedding Predictive Architectures (LeCun, 2022) learn latent predictions of future or masked content. I-JEPA (Assran et al., 2023), V-JEPA (Bardes et al., 2024), and Image World Models (Garrido et al., 2024) demonstrate strong latent spaces but do not primarily study whether spatial factors compose under held-out bindings. Recent work decomposes JEPA latent spaces into subspaces or sparse components for improved structure (Thil et al., 2026; Kuang et al., 2026), but does not evaluate compositional generalization of spatial factors. We use frozen DINOv3 and SigLIP2 vision transformers as representations and ask what readout

makes their latent innovation fields compositionally recoverable under a specified intervention and supervision protocol.

Object-centric and slot models. MONet (Burgess et al., 2019), IODINE (Greff et al., 2019), Slot Attention (Locatello et al., 2020), and its video extensions (Kipf et al., 2022; Singh et al., 2022) decompose scenes into object slots, capturing the *where* axis of our framework. A slot carries the object together with whatever happens to it; we model the operation as its own separately routed axis. Tensor-product representations (Smolensky, 1990) bind roles to fillers via outer products; our support×operation factorization is analogous, with spatial masks as role selectors and operation posteriors as fillers.

Probing and concept analysis. Linear probes (Alain & Bengio, 2017; Belinkov, 2022), concept activation vectors (Kim et al., 2018), and control tasks (Hewitt & Liang, 2019) read out properties from frozen representations but typically test one factor at a time; Network Dissection (Bau et al., 2017) localizes concepts to units. Our readout is a structured probe that reads out *interventions* (the innovation field) under a *compositional* protocol, and is scored on a held-out support×operation binding rather than on single-attribute classification.

Compositional generalization. Systematic generalization tests (Lake & Baroni, 2018; Keysers et al., 2020) evaluate whether a model handles unseen compositions of known primitives. Wiedemer et al. (2023) provide theoretical conditions for compositional generalization from first principles. Compositional zero-shot learning (CZSL) evaluates unseen attribute–object pairs (Misra et al., 2017); our support×operation grid is structurally analogous, with the difference that we evaluate a readout atop a frozen encoder rather than an end-to-end classifier. Our leave-one-cell-out protocol instantiates systematic generalization for spatial-visual factors.

Causal representation learning. Causal representation learning (Schölkopf et al., 2021; von Kügelgen et al., 2021) and nonlinear ICA (Hyvärinen et al., 2019) study whether latent variables can be identified from interventions or auxiliary signals; their evaluation criterion is identifiability of individual coordinates. We address a complementary question: given a frozen encoder with known intervention structure, can a readout *compose* spatial bindings that were absent from training? Our setting is structurally simpler (no identifiability proof is attempted), but adds a compositional held-out-binding evaluation that identifiability analyses do not require.