REPRESENTATIONAL SIMILARITY ANALYSIS: COUNTERING INCOMPLETENESS WITH FUNCTIONAL NEURONAL ORGANIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Stimulus representations are widely used in neuroscience and machine learning to compare neural systems, such as different brain regions, organisms, and deep learning models. These representations, analyzed with decoding (perceptual) manifolds and alignment metrics such as Representational Similarity Analysis (RSA), is commonly taken as evidence of similar computation. This approach has a fundamental weakness: it is misleading to assume that representational geometry reflects the neuronal population as a whole, when such representations may be shaped by a small subset of neurons. We show that the complementary *encoding* paradigm addresses this: it characterizes how neurons are organized globally in terms of their responses, providing insight into how the stimulus representation is implemented by neurons within a population. We demonstrate across experiments in biological systems and deep learning models that high representational alignment can arise from small, non-representative subpopulations of neurons. Further, alignment metrics are insensitive to encoding manifold topology (how function is distributed across neurons), despite this being a key signature of differentiation across biological systems. A controlled MNIST experiment provides causal evidence: alignment metrics change only marginally even when encoding topology is causally manipulated. Overall, similarity in stimulus representations, as measured by classic alignment metrics, does not imply similarity in function or computation, motivating the use of encoding manifolds as a complementary tool for comparing neural systems.

1 INTRODUCTION

A central goal in systems neuroscience is to compare computations across neural systems — brain regions, organisms, or deep learning models (Kriegeskorte et al., 2008; Khaligh-Razavi & Kriegeskorte, 2014; Schrimpf et al., 2020). The dominant approach probes each system with a shared set of stimuli and records the population response for each stimulus (Chung et al., 2018; Bertram et al., 2025; DiCarlo et al., 2012). These responses can be analyzed by decoding the stimuli from the activity (Kay et al., 2008; Mathis et al., 2024). This decoding approach asks how stimuli are represented in the model, that is, how the model behaves on these stimuli. These representations are analyzed using decoding (perceptual) manifolds and scalar alignment scores such as Representational Similarity Analysis (RSA) (Kriegeskorte et al., 2008) and Centered Kernel Alignment (CKA) (Kornblith et al., 2019). However, stimulus representations offer only a partial view of computational organization. How many neurons actually drive the observed representation, and how are they organized? We stress: Two systems that are not "digital twins" can achieve nearly identical alignment scores; the same decoding representation can be produced by qualitatively different internal functional organizations.

The reason is that the map from stimuli to representations is underspecified: many different internal organizations can produce the same population-level distance geometry (Ganguli & Sompolinsky, 2012; Gao & Ganguli, 2015; Bertram et al., 2026; Ghosh et al., 2025; Nellen et al., 2025). To claim functional similarity between two neural systems, one must look beyond the aggregate stimulus representations and examine *how* neurons are internally organized (Oota et al., 2023; Mathis et al., 2024). Inverse to decoding, the encoding approach answers this by asking how stimuli are encoded in the model's neurons. This yields the *encoding manifold* (Dyballa et al., 2024; Posani et al., 2025): rather than representing stimuli as points in neural-response space (the decoding or perceptual

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.

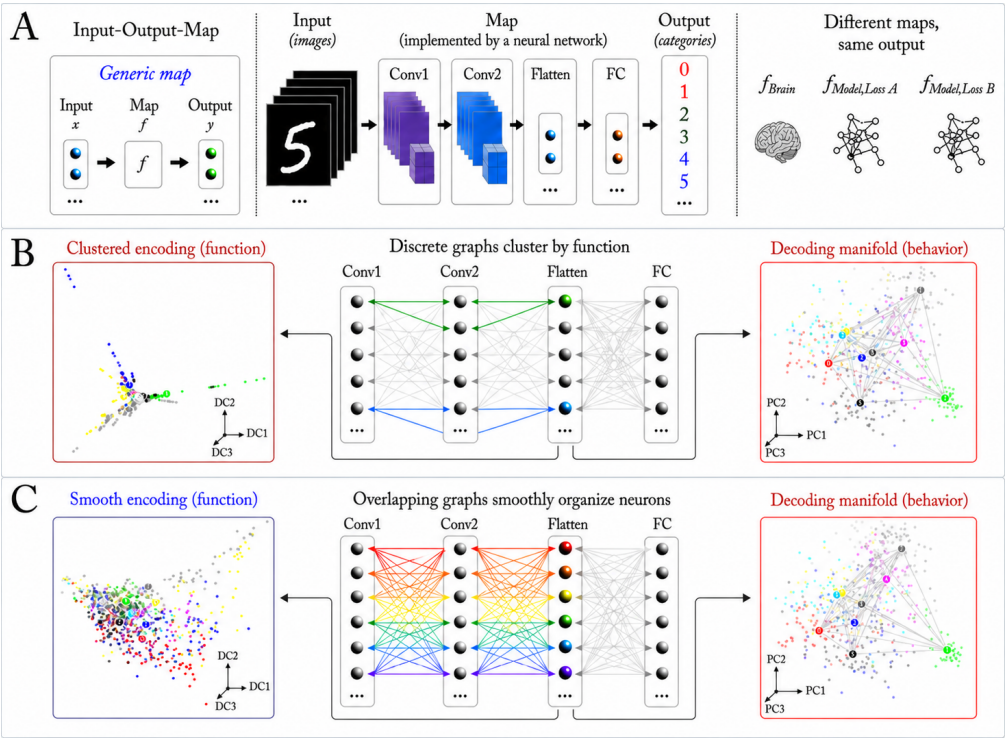

Figure 1: **Encoding and decoding manifolds provide complementary views of neural systems**. (A) A neural network as an input-output map. The same behavior – categorization – can be achieved with different maps. (B) Stimuli can be represented in neural coordinates (decoding manifold, right, colors represent different stimuli) or neurons can be represented in stimulus coordinates (encoding manifold, left, colors represent neurons' preferred stimulus). A clustered decoding manifold implies discrimination between stimulus classes; a clustered encoding manifold (non-overlapping preferred stimulus 'arms') implies separation between neurons' response patterns. (C) A similar decoding topology (compare with B) can be achieved by a different network, with neurons that exhibit a continuous encoding topology (left), which suggests a less sparse computational graph (see Sec. 4.4).

manifold), the encoding manifold represents *neurons* as points in stimulus-response space: nearby neurons are similarly tuned, just as nearby stimuli on the decoding manifold share similar population responses. The global topology of this neural point cloud — whether it forms discrete clusters, as in retinal ganglion cell types, or a smooth continuum, as in mouse V1 — directly reflects the functional architecture of the population (Figure 1).

Decoding metrics treat the neural population as a monolithic unit and return a single scalar. This paper reveals a key structural limitation: *the same scalar summary of decoding behavior (RSA, CKA, or classification accuracy) can be achieved by functionally heterogeneous populations*. Specifically:

1. A small, selective subset of neurons can replicate the full population's decoding performance and may achieve this through *different* neural subpopulations.
2. Representational alignment metrics, which operate on the same population-level distance matrices, inherit this blind spot.
3. Encoding and decoding manifold topology can be independently manipulated, providing *causal* evidence that decoding metrics are insensitive to internal functional organization.

We demonstrate these findings across mouse retina and V1 (Dyballa et al., 2024), five cortical areas of the Allen Brain Observatory (de Vries et al., 2020), and several machine learning models (Lappalainen et al., 2024; Tran et al., 2018) (Appendix B.1). These findings have direct practical consequences: a model could be declared a faithful replica of cortex based on matched RSA or CKA while its internal organization is structurally distinct. We therefore propose using encoding manifolds, and Gromov–Wasserstein distance as a quantitative measure of encoding manifold coverage, as complementary diagnostics for any decoding-based similarity claim (Dyballa et al., 2024; Bertram et al., 2026).

2 Related Work

RSA (Kriegeskorte et al., 2008) and CKA (Kornblith et al., 2019) are central tools for the comparison of neural representations. However, they operate on population-level summaries of activity, obscuring the internal structure and functional heterogeneity of the underlying neurons. Thus, researchers have questioned their statistical and theoretical properties (Ding et al., 2021; Ahlert et al., 2024; Bo et al., 2025; Höfling et al., 2026; Murphy et al., 2024; Harvey et al., 2024; Almudévar & Ortega, 2026; Avitan & Golan, 2025; Feather et al., 2025; Kriegeskorte & Douglas, 2019; Diedrichsen & Kriegeskorte, 2017). Related work Davari et al. (2022); Harvey et al. (2024); Bo et al. (2025) has proposed comparing systems via alignment of their output functions. In the interpretability literature, circuit tracing asks a similar question by investigating the function implemented by a neural network (Ameisen et al., 2025; Lindsey et al., 2024; Dunefsky et al., 2024). Our work is complementary: we shift the focus from output alignment and task performance to internal organization, analyzing functional topology within neural populations to reveal differences not captured by traditional similarity metrics.

RSA, CKA, and Procrustes are invariant to orthogonal transforms of the neuron axis, so their insensitivity to neuron-level organization is expected in principle (Kriegeskorte & Wei, 2021); we show that this gap is large in real data and make it measurable. Most closely related, Khosla & Williams (2024) propose the Soft-Matching Distance (SMD), a Wasserstein-style optimal-transport metric between tuning vectors in a shared stimulus frame. Our Gromov-Wasserstein-based encoding manifold similarity (Section 3.2) has two advantages over SMD. First, GW needs no shared stimulus frame: its transport cost is built entirely from each population's own *within*-population (neuron-to-neuron) distances rather than from absolute tuning vectors in a common stimulus space, so it applies even without a shared stimulus battery or coordinate frame. Second, and more fundamentally, GW operates at a different level of description: rather than testing whether individual neurons match one-to-one (or via a soft coupling) across populations, it tests whether the *topology* of functional organization matches – how functional groups are arranged relative to one another – a more general property that is invariant to which specific neurons instantiate a given functional role. We compare both measures directly on the MNIST models in Appendix A.3.

3 Methods

Given a neural population response tensor $X \in \mathbb{R}^{N\times S\times K(\times T)}$ (neurons $\times$ stimuli $\times$ directions $\times$ time), the *decoding manifold* is the set of points traced out by the population activity vector as the stimulus varies. The time dimension may be collapsed depending on the underlying data or model. We define the decoding manifold as the collection $\mathcal{M} = \{x_{s,k}\}_{s,k} \subset \mathbb{R}^N$, described by the matrix $\hat{X} \in \mathbb{R}^{SK\times N}$, where $x_{s,k} \in \mathbb{R}^N$ is the time-averaged response vector for stimulus s and direction k. For visualization purposes (Cunningham & Yu, 2014), one may project $\hat{X}$ onto its top principal components, yielding an embedding $\mathbf{Z} = PCA_3(\hat{X}) \in \mathbb{R}^{SK\times 3}$ with axes PC1, PC2, PC3. The geometry of $\mathcal{M}$ and the clustering of stimuli on it is what decoding metrics (e.g., classification accuracy, RSA) quantify. Beyond time-averaged responses, one may also define *decoding trajectories*: for each (stimulus, direction) pair, the sequence $(X_{:,s,k,t})_{t=1}^{T}$ traces a path through $\mathbb{R}^N$. This is the space that temporal alignment metrics operate in.

3.1 Decoding Metrics

We evaluate eight complementary metrics organized into two groups: four *static* metrics that operate on time-averaged representations and four *trajectory* metrics that characterize temporal dynamics: k-NN accuracy, RSA, CKA, Procrustes R^2, time-resolved RSA, Speed Profile Correlation, Trajectory Procrustes R^2, and Dynamical Similarity Analysis. All metrics are normalized or correlation-based, which limits their dependence on subpopulation size, and high values mean higher alignment. Definitions and implementation details are given in Appendix B.2.1.

3.2 Encoding Manifold

The *encoding manifold* (Dyballa et al., 2024; Bertram et al., 2026) represents each neuron as a point in stimulus-response space, so that nearby neurons are similarly tuned; we denote this point cloud by $\mathcal{E}$. The global topology of $\mathcal{E}$ — in particular, whether neurons cluster into distinct functional

groups and how those groups are arranged — captures the internal organization of the population independently of its aggregate decoding performance.

Concretely, we construct the encoding manifold in three stages for each dataset, following Dyballa et al. (2024). First, the population response tensor is decomposed via Nonnegative Tensor Factorization (NTF) (Williams et al., 2018), yielding neural factors that embed each neuron into a stimulus-response space where proximity reflects similarity in tuning and temporal dynamics. The number of NTF factors is the main hyperparameter of this pipeline; we evaluate robustness to it and to sample size in Figure 9. Second, a weighted data graph over this neural encoding space is built using the Iterated Adaptive Neighborhoods (IAN) (Dyballa & Zucker, 2023) algorithm, which infers a locally adaptive similarity kernel without requiring a fixed neighborhood size. Third, diffusion maps (Coifman & Lafon, 2006; Coifman et al., 2005) are applied to this graph to obtain a low-dimensional embedding that preserves the intrinsic geometry of the population — the encoding manifold. We refer to the axes of this embedding as Diffusion Coordinates (DC1, DC2, DC3) throughout the paper, to distinguish them from the Principal Components (PC1, PC2, PC3) of the decoding manifold (Section 3). Full details are given in Appendix B.3.

Gromov–Wasserstein (GW) and related Optimal Transport (OT) distances (Peyré & Cuturi, 2020; Mémoli, 2011) have recently been applied to correspondence-free neural alignment: aligning fMRI activity maps across individuals (Thual et al., 2023), comparing noisy population dynamics across conditions (Nejatbakhsh et al., 2024), cross-session and cross-species alignment (Takeda et al., 2025), hierarchical multi-scale representational comparison (Shah & Khosla, 2025), and human–DNN perceptual correspondence (Takahashi et al., 2026).

Our use of GW is different: we apply it as an intrinsic measure of encoding manifold coverage, quantifying functional topology rather than representational alignment between systems (see Section 2 for a comparison to the Soft-Matching Distance). Intuitively, GW measures how well the internal pairwise-distance structure of one point cloud can be matched to that of another: two manifolds with the same shape score zero, while structurally incompatible ones score high. The idea is that if we can match neurons between two systems based on encoding position, then they are similar in the internal function they implement (see Appendix Figure 10 for an intuition of GW on synthetic data).

We compute normalized GW similarity between the two point clouds in diffusion coordinate space. Let $\mathbf{C}_{\mathcal{X}}$, $\mathbf{C}_{\mathcal{Y}}$ denote the pairwise Euclidean distance matrices in diffusion coordinates, and let μ, ν be uniform probability measures over $\mathcal{X}$ and $\mathcal{Y}$ respectively. Here $\Pi(\mu, \nu)$ denotes the set of all joint probability measures (couplings) on $\mathcal{X} \times \mathcal{Y}$ with marginals μ and ν, i.e. all valid soft matchings between the two point clouds, and π_{ij} is the mass transported from point $i \in \mathcal{X}$ to point $j \in \mathcal{Y}$. The GW distance and GW similarity are defined as:

$$\mathrm{GW}(\mathcal{X}, \mathcal{Y}) = \sqrt{\inf_{\pi \in \Pi(\mu,\nu)} \sum_{i,i',j,j'} |\mathbf{C}^{ii'}_{\mathcal{X}} - \mathbf{C}^{jj'}_{\mathcal{Y}}|^2 \, \pi_{ij} \pi_{i'j'}}, \qquad \mathrm{GW}_{\mathrm{Sim}}(\mathcal{E}_{\mathrm{sub}}) = 1 - \frac{\mathrm{GW}(\mathcal{E}_{\mathrm{sub}}, \mathcal{E}_{\mathrm{full}})}{\mathrm{GW}_{\max}},$$

where $\mathcal{E}_{\mathrm{full}}$ is the full population's encoding manifold, $\mathcal{E}_{\mathrm{sub}} \subset \mathcal{E}_{\mathrm{full}}$ the points of the subpopulation, and $\mathrm{GW}_{\max} = \mathrm{GW}(\mathcal{E}_{\mathrm{full}}, \mathcal{E}_{\mathrm{coll}})$, where $\mathcal{E}_{\mathrm{coll}}$ collapses all points to their centroid, thereby providing a baseline for the maximal distance all points could travel ($\epsilon = 10^{-8}$ jitter to prevent degeneracy).

4 Results

4.1 Decoding Metrics Saturate with Small Subpopulations

First we show that decoding scores can be reproduced by a small fraction of neurons. We perform a population-size sweep from the full population down to a single neuron under 11 neuron-selection strategies (random, and high/low curvature, stability, classifiability, orientation selectivity index, and Principal Component (PC) contribution; Appendix B.5). Figure 2 shows results for Retina, V1, and VISp for gratings and natural videos. Across biological systems, metrics reach near-ceiling values with 5% of neurons or fewer under information-maximizing selection strategies (i.e. picking high-scoring neurons, e.g. high OSI or curvature, rather than the low-scoring variant). RSA and CKA remain near ceiling under random selection at moderate fractions. Proc. R^2 and trajectory metrics produce lower alignment scores only below 1% of neurons.

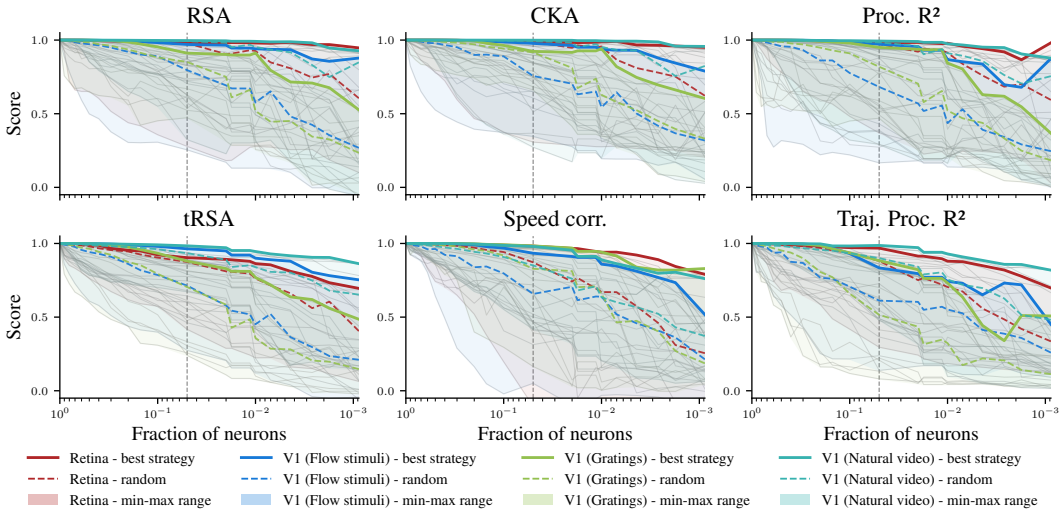

Figure 2: Population-size sweep across Retina, V1 (flows, gratings, natural) for six decoding metrics (more in Figure 6). Solid lines: best-performing selection strategy; dashed lines: random selection; shaded bands: min–max range across all strategies. The vertical dashed line marks 5% of neurons (used in Figure 3). Static metrics plateau near ceiling at 5% or below under the best strategy.

Decoding metrics also vary more across the 11 selection strategies than GW does: across fractions, the standard deviation of GW similarity across strategies is roughly 2.2–2.6× smaller than the mean standard deviation of the six decoding metrics (Table 1), i.e. encoding alignment is comparatively insensitive to *how* the subpopulation is selected, while decoding alignment is not.

This raises a fundamental question: if decoding metrics can be matched by a handful of neurons, what functional role do the remaining neurons play? If these metrics truly captured functional alignment, one would expect any small subpopulation to be an equally faithful or unfaithful representative of the full population. The alternative — that most neurons implement functions invisible to decoding metrics — is what the following sections test directly.

4.2 Encoding Manifold Position Determines Decoding Fidelity

To link subpopulation decoding fidelity to encoding manifold coverage, in Figure 3 we select size-matched subpopulations by maximizing (top row, A/C/E/G) or minimizing (bottom row, B/D/F/H) each neuron's contribution to the first principal component of the decoding space — yielding near best- and worst-case samples for decoding metrics, respectively (Appendix B.6).

The bar chart confirms this selection produces a wide range of decoding scores: high-PC1 subpopulations largely recover the full-population profile across all eight metrics, while low-PC1 subpopulations yield substantially lower scores. The encoding manifold panels tell the complementary story: regardless of whether a subpopulation scores well on decoding metrics, it occupies only a small, localized region of the encoding manifold. This holds for all four systems and at 1% and 10% (Figures 7, 8). Where decoding metrics capture differences between the high- and low-PC1 conditions, GW does too: high-PC1 subpopulations are significantly closer to the full population in GW self-similarity (Figure 3 J), though both fall below size-matched random subsets (0.66 and 0.56 vs. 0.77).

Table 1: Standard deviation across the 11 selection strategies, by fraction of neurons kept.

	RSA	CKA	Proc	tRSA	Speed	tProc	**GW (ours)**	ratio (mean of 6 / GW)
1%	0.243	0.240	0.274	0.236	0.317	0.266	0.100	2.6×
10%	0.168	0.171	0.221	0.164	0.225	0.207	0.089	2.2×

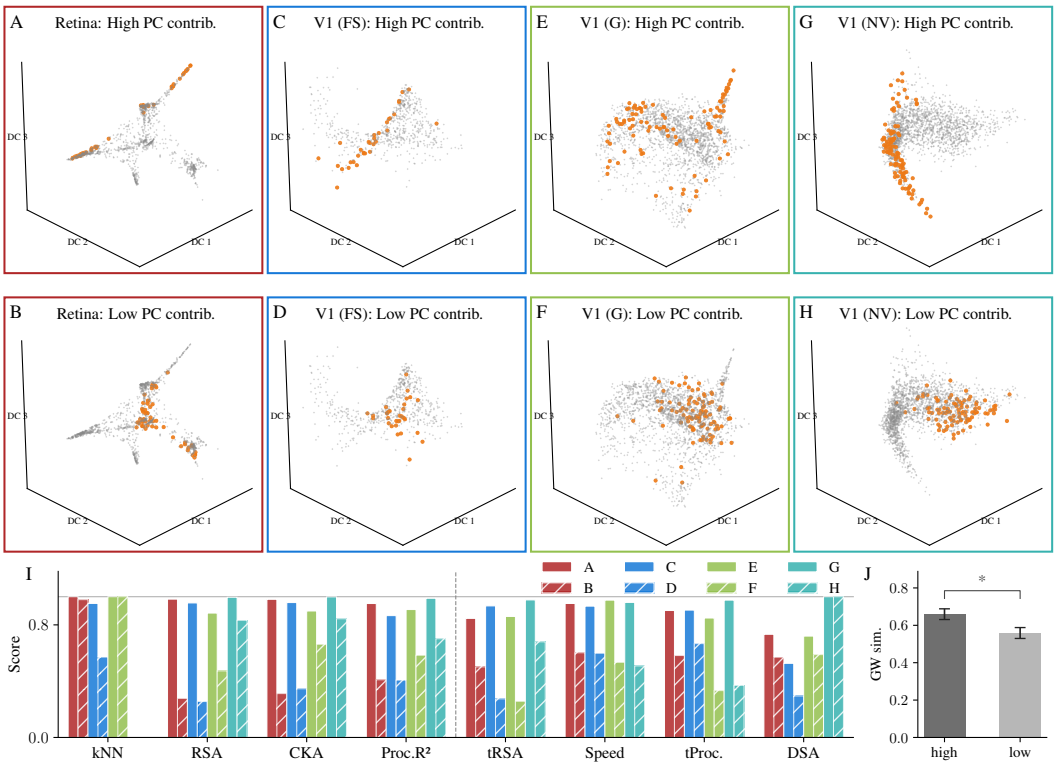

Figure 3: Encoding manifold coverage and decoding scores for 5% subpopulations selected by highest (top row, solid bars) or lowest (bottom row, hatched bars) first-PC contribution in decoding space, for Retina, V1, VISp (gratings), and VISp (natural). Orange dots show selected neurons on the encoding manifold (gray: full population); in all cases the selection covers a small, localized region of the manifold. Bar chart shows all eight metrics; RSA and CKA show the largest differences between conditions. Robustness across sizes in Figures 7 (1%) and 8 (10%) (Wilcoxon signed-rank; * $p<0.05$).

Encoding manifold topology characterizes function. However, the encoding manifold, by design, organizes the full population of neurons by function, and has been validated against known results from biology. For example, the retinal encoding manifold clusters neurons by known retinal ganglion cell types (Dyballa et al., 2024). Consistent with the absence of known functional groups of neurons in mouse V1 (Dyballa et al., 2024; Nellen et al., 2025), the V1 encoding manifold is continuous. These stable biological properties of computation are robustly captured by the encoding manifold pipeline (Appendix B.4, Figure 9). Therefore, we argue that the encoding manifold's coverage and topology are relevant properties for functional alignment. Our artificial MNIST example provides additional evidence for these claims in a controlled setting (Section 4.4). As a result, sampling region-constrained subpopulations on the encoding manifold omits biologically relevant functions, such as most retinal ganglion cell types. Still, the high-PC1 subpopulations yield aligned decoding representations, dissociating functional from representational alignment. The high/low-PC1 contrast is driven by response amplitude, not tuning selectivity or a coarse cell-type split (Appendix A.1).

4.3 Encoding Manifold Topology Can Be Disrupted While Preserving Decoding Metrics

Having established encoding manifold topology as a description of a system's function, we now manipulate it. We use *Farthest Point Sampling* (FPS) to construct subpopulations (Appendix B.8): n seed neurons are selected by maximally covering the encoding manifold embedding, and for each seed its m nearest neighbors in the manifold are included. By varying n (breadth of coverage) and m (local neighborhood density), we can interpolate between a **continuous**, topologically smooth but sparse sample ($m = 1$, many seed points) and a more locally **clustered** sample ($m > 1$). Figure 4 shows results for Retina, V1, and Allen V1 (VISp).

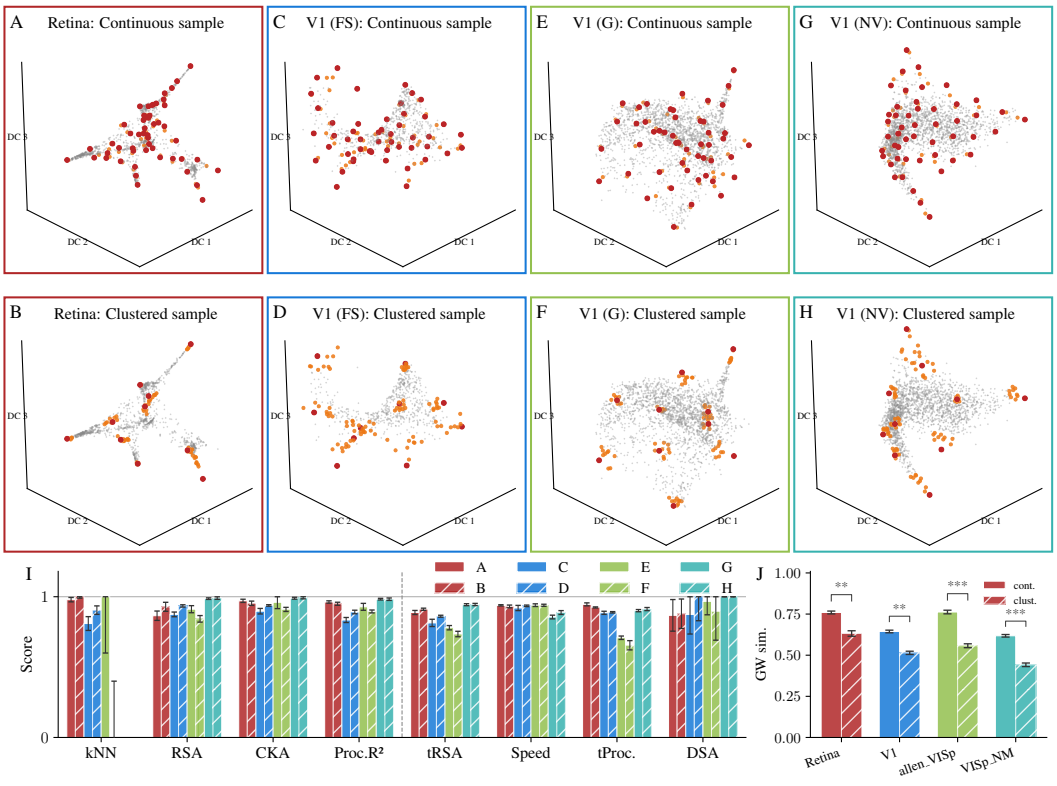

Figure 4: FPS-based subpopulations with continuous (top row, dark red) vs. clustered (bottom row, orange) encoding manifold topology for Retina, V1, VISp (gratings), and VISp (natural). Continuous subpopulations are sparsely distributed across the full manifold; clustered subpopulations are concentrated in local neighborhoods. Bar chart shows all eight metrics for both conditions; error bars are 1 s.d. over FPS seeds. Scores are similar for both topologies across nearly all systems and metrics, showing that decoding metrics are insensitive to this difference in encoding structure. (J: mean $\pm$ SEM, paired t-test over the 5 FPS seeds; ** $p<0.01$, *** $p<0.001$)

All decoding metrics remain high for both functional topologies across systems (Figure 4). Differences between conditions are small and within error bars for the majority of systems and metrics. Only the Procrustes (static and temporal) scores show some variation for a subset of the datasets. Together, Retina, V1, and ten Allen cortical datasets (five areas $\times$ two stimulus classes) establish that the insensitivity of RSA, CKA, DSA and other decoding metrics to encoding manifold topology is robust across systems and stimuli. Machine learning models (FNN, Flyvision, R(2+1)D, ViT, Raptor) show a similar pattern, with larger gaps for FNN and ViT (Appendix A.2.2). Per-dataset scores and manifold visualizations for all 17 datasets are given in Table 7 and Figures 11–27.

4.4 Causal Dissociation via Controlled Training

The experiments above show *correlationally* that decoding metrics are insensitive to encoding manifold topology. To establish this as a *causal* property of the metrics, we directly manipulate encoding topology by design. We train a three-layer CNN on MNIST digit classification (10 classes, $\approx$ 60 000 images; details in Appendix B.9) and add an auxiliary loss $L_{\text{cluster}}(\lambda)$ with strength λ that progressively pulls together hidden-layer neurons with the same preferred digit, systematically clustering the encoding manifold. Task accuracy is held approximately constant ($\approx 98.5\%$) across all values of λ.

Table 5 reports decoding metrics (RSA, CKA, Procrustes R^2) and encoding similarity (GW between the IAN encoding manifold of the trained model and the baseline) as λ increases from 0 to 50. Over this range, GW similarity to the baseline drops from 1.00 to 0.28 — a fundamental topological change in which the previously continuous population response space becomes sharply clustered by preferred

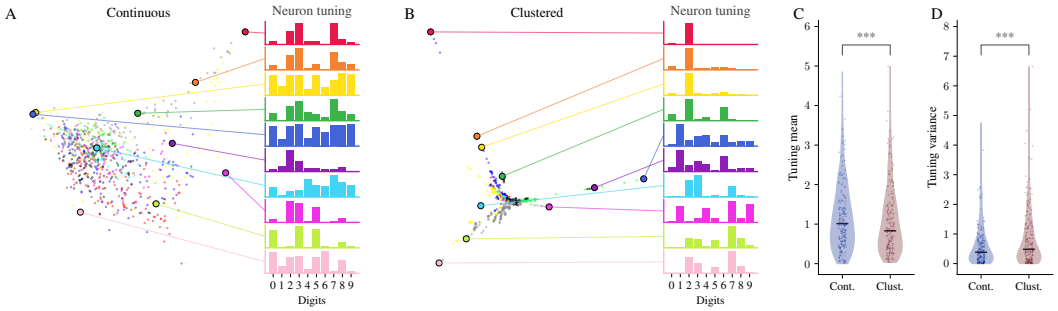

Figure 5: MNIST experiment: continuous (A) vs. clustered (B) encoding manifold colored by neurons' preferred stimuli. Tuning histograms show graded, multi-class responses in A and sharp, single-class peaks in B. Neurons in the clustered model have significantly lower tuning mean (C) and higher variance (D), indicating sharper class-selective responses. (Wilcoxon rank-sum; *** $p<0.001$)

digit (Figure 5). We call the baseline encoding manifold (Figure 5 A) continuous since neurons with preferred stimuli overlap and are smoothly scattered across the manifold. On the other hand, the clustered encoding manifold (Figure 5 B) shows clear preferred stimulus clusters that are pushed away from other neurons, resulting in the preferred stimulus 'arms' extending into different directions on the manifold. Up to $\lambda = 10$, RSA, CKA, and Procrustes R^2 remain above 0.93 and task accuracy is unchanged.

The dissociation is not a ceiling effect: at $\lambda = 10$, where decoding metrics are near ceiling, GW similarity is 0.502 (Table 2), indicating a substantially reorganized internal structure. Thus, the insensitivity of decoding metrics to encoding topology is a structural property of the metrics themselves, not of the specific biological or neural network data analyzed above.

Functional Differences. Despite matched decoding behavior, the two models are functionally distinct beyond encoding manifold topology. First, neurons in the clustered model show significantly lower mean tuning and higher tuning variance (Figure 5 C, D, $p<0.001$), reflecting sharper, class-selective responses (Figure 5 B) rather than graded, multi-class tuning (Figure 5 A). Second, while the models perform similarly on in-distribution data, the clustered model is slightly less robust to strong out-of-distribution Gaussian blur (Table 2; $p<0.05$ pooled over $\sigma_{\text{blur}} \geq 4$, Table 6). Third, gradient-activation attribution analysis (Table 2; details in Appendix B.9.8) reveals that the constrained model uses more class-specific, less shared computational circuits: attribution entropy (both layers) and FC overlap are significantly lower ($p<0.001$), and weight-graph assortativity is significantly higher, indicating that same-class neurons have more similar weights. Thus, although their RSA, CKA, and

Table 2: MNIST results (n=20 seeds, mean ± SEM). **Top:** decoding and encoding metrics for λ=10 vs. baseline; baseline column compares pairs of baseline seeds. **Bottom:** functional organization metrics for λ=10 vs. baseline (Wilcoxon signed-rank for blur, rank-sum otherwise; Holm-corrected; * $p<0.05$, *** $p<0.001$.)

Metric	Baseline (λ=0)	Clustered (λ=10)	Sig.
RSA	0.958 ± 0.001	0.962 ± 0.003	ns
CKA	0.976 ± 0.001	0.976 ± 0.002	ns
Proc. R^2	0.950 ± 0.002	0.943 ± 0.010	ns
GW Sim (ours)	**0.958** ± 0.001	**0.502** ± 0.022	***
Functional Differences			
Blur robustness, mean acc. ($\sigma_{\text{blur}} \geq 4$)	0.330 ± 0.011	0.284 ± 0.013	*
FC Attribution Entropy	5.919 ± 0.008	5.756 ± 0.007	***
FC Attribution Overlap	0.463 ± 0.002	0.407 ± 0.005	***
FC Assortativity	0.077 ± 0.001	0.086 ± 0.002	*
Conv2 Attribution Entropy	3.232 ± 0.008	3.097 ± 0.019	***
Conv2 Attribution Overlap	0.902 ± 0.003	0.889 ± 0.006	ns
Conv2 Assortativity	0.277 ± 0.035	0.501 ± 0.030	***

Procrustes differences are small (Appendix A.3), the two models implement their shared behavior through different internal functional architectures.

5 DISCUSSION

Decoding metrics fail to capture functional diversity within populations. The population-size sweep admits two interpretations: either neurons are highly redundant, or most neurons implement functions that decoding metrics do not capture. The region-sampling experiment (Section 4.2) argues against full redundancy: if neurons were interchangeable, any size-matched subpopulation should perform equivalently, but this is not what we observe. Only some subpopulations reproduce full-population behavior, and these occupy a small region of the encoding manifold. As this contrast tracks response amplitude (Appendix A.1), RSA and CKA with Euclidean distances are dominated by high-amplitude neurons.

Encoding manifold topology is invisible to decoding metrics. The FPS experiment (Section 4.3) isolates this effect: for size-matched subpopulations with continuous, distributed vs. locally clustered coverage of the encoding manifold, differences under standard decoding metrics are small and inconsistent in sign, with RSA, CKA, and Procrustes R^2 remaining high across systems despite clear topological differences.

Implications for model–brain comparisons: behavioral *vs.* functional alignment. Current evaluations of neural models using RSA, CKA, or decoding accuracy establish *behavioral* alignment—what the population does, collectively—but not *functional* alignment—how that behavior is implemented. A model may match biological responses while implementing a fundamentally different internal organization: distinct functional subpopulations, altered inter-group relationships, and different representational roles distributed across neurons. The MNIST experiment makes this concrete: models matched on decoding metrics differ significantly in tuning sharpness, out-of-distribution robustness, and the structure of their computational circuits (Section 4.4). We therefore recommend complementing decoding-based comparisons with encoding manifolds and GW-based measures of functional organization.

Limitations. First, the MNIST experiment uses a specific clustering loss and a relatively shallow architecture; whether the same decoupling holds for deeper networks, other training objectives, or naturally trained models remains open. Second, the biological analyses are correlational: while we observe that decoding metrics fail to distinguish functionally distinct subpopulations, encoding topology cannot be directly manipulated *in vivo*. Third, decoding metrics compare each subpopulation to the full population containing it, which may favor high-amplitude subsets. Because GW compares populations only through their within-population (neuron-to-neuron) distances, two populations with matched internal geometry but tuned to unrelated stimulus content could, in principle, score near-zero GW distance. While this is unlikely in a real application, it is the complementary failure mode to Soft-Matching Distance (Khosla & Williams, 2024), which avoids it by comparing tuning vectors directly but for the same reason requires both populations to share a stimulus frame. This failure mode can only produce spurious matches (too small a GW distance), so it cannot explain the GW differences we report.

Conclusion. A small, non-representative subpopulation can reproduce the full population's alignment metrics, and these metrics remain insensitive to fundamental differences in encoding topology. This is not a limitation of small sample sizes or noisy data, but a structural property of population-level distance measures. Alignment metrics measure what a population *does* under a fixed stimulus set; the encoding manifold captures *how* it does it. A population that scores the same on alignment metrics may be implementing the same behavior through an entirely different internal organization.

Our results do not invalidate decoding-based similarity metrics; rather, they delimit what those metrics can support. RSA, CKA, and related measures are informative about how well a population can be read out under a given stimulus set, but they do not, by themselves, determine how that population is internally organized. Encoding manifolds provide a complementary description of that organization, and GW offers a principled way to compare them. In this sense, the appropriate unit of comparison is not a single scalar alignment score, but a pair of measurements: what a system does, and how it is organized to do it.

AI USE STATEMENT

In this work, we used generative AI tools (large language models, primarily Anthropic's Claude via the Claude Code coding assistant) to aid and polish writing, for retrieval and discovery of related work, and for research ideation and execution. We have not used generative AI tools to prove mathematical claims. All neural recordings are published experimental data, and all model activations were obtained by running the cited models on our stimuli.

Writing. LLMs were used to propose paragraphs and edit author-written text for clarity, grammar, and concision. All scientific content, arguments, and conclusions were written by the authors, and every suggested edit was reviewed before being accepted.

Retrieval and discovery. LLMs were used to search for and summarize potentially relevant literature. Every cited work was read by the authors, and all citations and bibliographic details were checked against the original sources.

Research ideation and execution. LLMs were used to discuss analysis ideas, propose control experiments, and critique our interpretations. The research question, framing, and choice of analyses are the authors'. LLM coding assistants also wrote and refactored parts of the analysis and plotting code, including appendix synthetic dataset generation code. All AI-generated code was reviewed by the authors, and each analysis pipeline was validated on cases with a known answer (e.g., synthetic manifolds with controlled topology and null/shuffle controls) before being applied to the data. All reported numbers were produced by code the authors ran and inspected.

We have reviewed all AI-assisted work, and we take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work analyzes previously collected biological recordings (mouse retina and V1, and the Allen Brain Observatory dataset) and does not involve any new data collection from human or animal subjects. All animal procedures underlying these datasets were approved under the ethics protocols of their original publications/sources; we did not perform any additional animal experiments. We do not foresee other ethical concerns (e.g. dual-use, privacy, or fairness issues) arising from this work.

REPRODUCIBILITY STATEMENT

All random seeds are fixed and reported, and code is made available (Appendix B.12); implementation and hyperparameter details for every metric and pipeline are given in Appendix B.2.1–B.11.

REFERENCES

Jannis Ahlert, Thomas Klein, Felix Wichmann, and Robert Geirhos. How Aligned are Different Alignment Metrics?, 2024. URL http://arxiv.org/abs/2407.07530.

Antonio Almudévar and Alfonso Ortega. Bridging Functional and Representational Similarity via Usable Information, 2026. URL http://arxiv.org/abs/2601.21568.

Emmanuel Ameisen, Maheen Syed, Jack Lindsey, Alex Tamkin, Adly Templeton, Jonathan Marcus, Thomas Conerly, Joshua Batson, and Christopher Olah. Circuit tracing: Revealing computational graphs in language models. *Transformer Circuits Thread*, 2025. URL https://transformer-circuits.pub/2025/attribution-graphs/methods.html.

Itamar Avitan and Tal Golan. Model-Behavior Alignment under Flexible Evaluation: When the Best-Fitting Model Isn't the Right One, 2025. URL http://arxiv.org/abs/2510.23321.

Brett W. Bader, Tamara G. Kolda, et al. Tensor toolbox for matlab, version 3.6. https://www.tensortoolbox.org, 2023.

Johannes Bertram, Luciano Dyballa, T. Anderson Keller, Savik Kinger, and Steven W. Zucker. Manifolds and Modules: How Function Develops in a Neural Foundation Model, 2025. URL http://arxiv.org/abs/2512.07869.

Johannes Bertram, Luciano Dyballa, Anderson Keller, Savik Kinger, and Steven W. Zucker. How 'Neural' is a Neural Foundation Model?, 2026. URL http://arxiv.org/abs/2601.21508.

Yiqing Bo, Ansh Soni, Sudhanshu Srivastava, and Meenakshi Khosla. Evaluating Representational Similarity Measures from the Lens of Functional Correspondence, 2025. URL http://arxiv.org/abs/2411.14633.

SueYeon Chung, Daniel D. Lee, and Haim Sompolinsky. Classification and Geometry of General Perceptual Manifolds. 8(3):031003, 2018. ISSN 2160-3308. doi: 10.1103/PhysRevX.8.031003. URL http://arxiv.org/abs/1710.06487.

Ronald R. Coifman and Stéphane Lafon. Diffusion maps. 21(1):5–30, 2006. ISSN 1063-5203. doi: 10.1016/j.acha.2006.04.006. URL https://www.sciencedirect.com/science/article/pii/S1063520306000546.

Ronald R Coifman, Stephane Lafon, Ann B Lee, Mauro Maggioni, Boaz Nadler, Frederick Warner, and Steven W Zucker. Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps. *Proceedings of the national academy of sciences*, 102(21):7426–7431, 2005.

John P Cunningham and Byron M Yu. Dimensionality reduction for large-scale neural recordings. *Nature neuroscience*, 17(11):1500–1509, 2014.

MohammadReza Davari, Stefan Horoi, Amine Natik, Guillaume Lajoie, Guy Wolf, and Eugene Belilovsky. Reliability of CKA as a Similarity Measure in Deep Learning, 2022. URL http://arxiv.org/abs/2210.16156.

Saskia E. J. de Vries, Jerome A. Lecoq, Michael A. Buice, Peter A. Groblewski, Gabriel K. Ocker, Michael Oliver, David Feng, Nicholas Cain, Peter Ledochowitsch, Daniel Millman, Kate Roll, Marina Garrett, Tom Keenan, Leonard Kuan, Stefan Mihalas, Shawn Olsen, Carol Thompson, Wayne Wakeman, Jack Waters, Derric Williams, Chris Barber, Nathan Berbesque, Brandon Blanchard, Nicholas Bowles, Shiella D. Caldejon, Linzy Casal, Andrew Cho, Sissy Cross, Chinh Dang, Tim Dolbeare, Melise Edwards, John Galbraith, Nathalie Gaudreault, Terri L. Gilbert, Fiona Griffin, Perry Hargrave, Robert Howard, Lawrence Huang, Sean Jewell, Nika Keller, Ulf Knoblich, Josh D. Larkin, Rachael Larsen, Chris Lau, Eric Lee, Felix Lee, Arielle Leon, Lu Li, Fuhui Long, Jennifer Luviano, Kyla Mace, Thuyanh Nguyen, Jed Perkins, Miranda Robertson, Sam Seid, Eric Shea-Brown, Jianghong Shi, Nathan Sjoquist, Cliff Slaughterbeck, David Sullivan, Ryan Valenza, Casey White, Ali Williford, Daniela M. Witten, Jun Zhuang, Hongkui Zeng, Colin Farrell, Lydia Ng, Amy Bernard, John W. Phillips, R. Clay Reid, and Christof Koch. A large-scale standardized physiological survey reveals functional organization of the mouse visual cortex. 23(1):138–151, 2020. ISSN 1546-1726. doi: 10.1038/s41593-019-0550-9. URL https://www.nature.com/articles/s41593-019-0550-9.

James J DiCarlo, Davide Zoccolan, and Nicole C Rust. How does the brain solve visual object recognition? *Neuron*, 73(3):415–434, 2012.

Jörn Diedrichsen and Nikolaus Kriegeskorte. Representational models: A common framework for understanding encoding, pattern-component, and representational-similarity analysis. *PLoS Computational Biology*, 13(4):e1005508, 2017. doi: 10.1371/journal.pcbi.1005508.

Frances Ding, Jean-Stanislas Denain, and Jacob Steinhardt. Grounding Representation Similarity with Statistical Testing, 2021. URL http://arxiv.org/abs/2108.01661.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, 2021. URL http://arxiv.org/abs/2010.11929.

Jacob Dunefsky, Philippe Chlenski, and Neel Nanda. Transcoders Find Interpretable LLM Feature Circuits, November 2024.

Luciano Dyballa and Steven W. Zucker. IAN: Iterated Adaptive Neighborhoods for manifold learning and dimensionality estimation. 35(3):453–524, 2023. ISSN 0899-7667, 1530-888X. doi: 10.1162/neco_a_01566. URL http://arxiv.org/abs/2208.09123.

Luciano Dyballa, Andra M. Rudzite, Mahmood S. Hoseini, Mishek Thapa, Michael P. Stryker, Greg D. Field, and Steven W. Zucker. Population encoding of stimulus features along the visual hierarchy. 121(4):e2317773121, 2024. doi: 10.1073/pnas.2317773121. URL https://www.pnas.org/doi/10.1073/pnas.2317773121.

Jenelle Feather, Meenakshi Khosla, N. Apurva Ratan Murty, and Aran Nayebi. Brain-Model Evaluations Need the NeuroAI Turing Test, 2025. URL http://arxiv.org/abs/2502.16238.

Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Z. Alaya, Aurélie Boisbunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras, Nemo Fournier, Léo Gautheron, Nathalie T.H. Gayraud, Hicham Janati, Alain Rakotomamonjy, Ievgen Redko, Antoine Rolet, Antony Schutz, Vivien Seguy, Danica J. Sutherland, Romain Tavenard, Alexander Tong, and Titouan Vayer. POT: Python optimal transport. 22(1):78:3571–78:3578, 2021. ISSN 1532-4435. URL https://dl.acm.org/doi/10.5555/3546258.3546336.

Surya Ganguli and Haim Sompolinsky. Compressed sensing, sparsity, and dimensionality in neuronal information processing and data analysis. *Annual review of neuroscience*, 35:485–508, 2012.

Peiran Gao and Surya Ganguli. On simplicity and complexity in the brave new world of large-scale neuroscience. *Current opinion in neurobiology*, 32:148–155, 2015.

Arna Ghosh, Zahraa Chorghay, Shahab Bakhtiari, and Blake A. Richards. Why all roads don't lead to Rome: Representation geometry varies across the human visual cortical hierarchy, 2025. URL http://arxiv.org/abs/2509.13459.

Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, et al. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020. doi: 10.1038/s41586-020-2649-2. URL https://doi.org/10.1038/s41586-020-2649-2.

Sarah E. Harvey, David Lipshutz, and Alex H. Williams. What Representational Similarity Measures Imply about Decodable Information, 2024. URL http://arxiv.org/abs/2411.08197.

Larissa Höfling, Matthias Tangemann, Lotta Piefke, Susanne Keller, Katrin Franke, and Matthias Bethge. Only brains align with brains: Cross-region alignment patterns expose limits of normative models. *arXiv preprint arXiv:2604.21780*, 2026.

J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3): 90–95, 2007. doi: 10.1109/MCSE.2007.55.

Plotly Technologies Inc. Collaborative data science. https://plot.ly, 2015.

Mozes Jacobs, Thomas Fel, Richard Hakim, Alessandra Brondetta, Demba Ba, and T. Andy Keller. Block-Recurrent Dynamics in Vision Transformers, 2026. URL http://arxiv.org/abs/2512.19941.

Kendrick N Kay, Thomas Naselaris, Ryan J Prenger, and Jack L Gallant. Identifying natural images from human brain activity. *Nature*, 452(7185):352–355, 2008.

Seyed-Mahdi Khaligh-Razavi and Nikolaus Kriegeskorte. Deep Supervised, but Not Unsupervised, Models May Explain IT Cortical Representation. 10(11):e1003915, 2014. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1003915. URL https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1003915.

Meenakshi Khosla and Alex H Williams. Soft matching distance: A metric on neural representations that captures single-neuron tuning. In Marco Fumero, Emanuele Rodolá, Clementine Domine, Francesco Locatello, Karolina Dziugaite, and Caron Mathilde (eds.), *Proceedings of UniReps: the First Workshop on Unifying Representations in Neural Models*, volume 243 of *Proceedings of Machine Learning Research*, pp. 326–341. PMLR, 15 Dec 2024. URL https://proceedings.mlr.press/v243/khosla24a.html.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised Contrastive Learning, March 2021.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of Neural Network Representations Revisited, 2019. URL http://arxiv.org/abs/1905.00414.

Nikolaus Kriegeskorte and Pamela K Douglas. Interpreting encoding and decoding models. *Current Opinion in Neurobiology*, 55:167–179, 2019. ISSN 0959-4388. doi: https://doi.org/10.1016/j.conb.2019.04.002. URL https://www.sciencedirect.com/science/article/pii/S0959438818301004. Machine Learning, Big Data, and Neuroscience.

Nikolaus Kriegeskorte and Xue-Xin Wei. Neural tuning and representational geometry. *Nature Reviews Neuroscience*, 22(11):703–718, 2021. doi: 10.1038/s41583-021-00502-3.

Nikolaus Kriegeskorte, Marieke Mur, and Peter A. Bandettini. Representational similarity analysis - connecting the branches of systems neuroscience. 2, 2008. ISSN 1662-5137. doi: 10.3389/neuro.06.004.2008. URL https://www.frontiersin.org/journals/systems-neuroscience/articles/10.3389/neuro.06.004.2008/full.

Nicholas Krämer et al. Tueplots, 2024. URL https://tueplots.readthedocs.io/en/latest/index.html.

Janne K. Lappalainen, Fabian D. Tschopp, Sridhama Prakhya, Mason McGill, Aljoscha Nern, Kazunori Shinomiya, Shin-ya Takemura, Eyal Gruntman, Jakob H. Macke, and Srinivas C. Turaga. Connectome-constrained networks predict neural activity across the fly visual system. 634(8036): 1132–1140, 2024. ISSN 1476-4687. doi: 10.1038/s41586-024-07939-3. URL https://www.nature.com/articles/s41586-024-07939-3.

Jack Lindsey, Adly Templeton, Jonathan Marcus, Thomas Conerly, Joshua Batson, and Christopher Olah. Sparse crosscoders for cross-layer features and model diffing. *Transformer Circuits Thread*, 2024. URL https://transformer-circuits.pub/2024/crosscoders/index.html.

Mackenzie Weygandt Mathis, Adriana Perez Rotondo, Edward F Chang, Andreas S Tolias, and Alexander Mathis. Decoding the brain: From neural representations to mechanistic models. *Cell*, 187(21):5814–5832, 2024.

Alex Murphy, Joel Zylberberg, and Alona Fyshe. Correcting Biased Centered Kernel Alignment Measures in Biological and Artificial Neural Networks, 2024. URL http://arxiv.org/abs/2405.01012.

Facundo Mémoli. Gromov–Wasserstein Distances and the Metric Approach to Object Matching. 11(4):417–487, 2011. ISSN 1615-3383. doi: 10.1007/s10208-011-9093-5. URL https://doi.org/10.1007/s10208-011-9093-5.

Amin Nejatbakhsh, Victor Geadah, Alex H. Williams, and David Lipshutz. Comparing noisy neural population dynamics using optimal transport distances, 2024. URL http://arxiv.org/abs/2412.14421.

Nina S. Nellen, Polina Turishcheva, Michaela Vystrčilová, Shashwat Sridhar, Tim Gollisch, Andreas S. Tolias, and Alexander S. Ecker. Learning to cluster neuronal function, 2025. URL http://arxiv.org/abs/2506.03293.

Subba Reddy Oota, Zijiao Chen, Manish Gupta, Raju S Bapi, Gael Jobard, Frédéric Alexandre, and Xavier Hinaut. Deep neural networks and brain alignment: Brain encoding and decoding (survey). *arXiv preprint arXiv:2307.10246*, 2023.

Mitchell Ostrow, Adam Eisen, Leo Kozachkov, and Ila Fiete. Beyond Geometry: Comparing the Temporal Structure of Computation in Neural Circuits with Dynamical Similarity Analysis, 2023. URL http://arxiv.org/abs/2306.10168.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, et al. Pytorch: An imperative style, high-performance deep learning library. *CoRR*, abs/1912.01703, 2019. URL http://arxiv.org/abs/1912.01703.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Gabriel Peyré and Marco Cuturi. Computational Optimal Transport, 2020. URL http://arxiv.org/abs/1803.00567.

Lorenzo Posani, Shuqi Wang, Samuel P Muscinelli, Liam Paninski, and Stefano Fusi. Rarely categorical, always high-dimensional: how the neural code changes along the cortical hierarchy. *biorxiv*, pp. 2024–11, 2025.

Martin Schrimpf, Jonas Kubilius, Ha Hong, Najib J. Majaj, Rishi Rajalingham, Elias B. Issa, Kohitij Kar, Pouya Bashivan, Jonathan Prescott-Roy, Franziska Geiger, Kailyn Schmidt, Daniel L. K. Yamins, and James J. DiCarlo. Brain-Score: Which Artificial Neural Network for Object Recognition is most Brain-Like?, 2020. URL https://www.biorxiv.org/content/10.1101/407007v2.

Shaan Shah and Meenakshi Khosla. Representational Alignment Across Model Layers and Brain Regions with Hierarchical Optimal Transport, 2025. URL http://arxiv.org/abs/2510.01706.

Soh Takahashi, Masaru Sasaki, Ken Takeda, and Masafumi Oizumi. Investigating Fine- and Coarse-grained Structural Correspondences Between Deep Neural Networks and Human Object Image Similarity Judgments Using Unsupervised Alignment. 195:108222, 2026. ISSN 08936080. doi: 10.1016/j.neunet.2025.108222. URL http://arxiv.org/abs/2505.16419.

Ken Takeda, Masaru Sasaki, Kota Abe, and Masafumi Oizumi. Unsupervised alignment in neuroscience: Introducing a toolbox for Gromov–Wasserstein optimal transport. 419:110443, 2025. ISSN 0165-0270. doi: 10.1016/j.jneumeth.2025.110443. URL https://www.sciencedirect.com/science/article/pii/S0165027025000846.

Alexis Thual, Huy Tran, Tatiana Zemskova, Nicolas Courty, Rémi Flamary, Stanislas Dehaene, and Bertrand Thirion. Aligning individual brains with Fused Unbalanced Gromov-Wasserstein, 2023. URL http://arxiv.org/abs/2206.09398.

Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A Closer Look at Spatiotemporal Convolutions for Action Recognition, 2018. URL http://arxiv.org/abs/1711.11248.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

Eric Y. Wang, Paul G. Fahey, Zhuokun Ding, Stelios Papadopoulos, Kayla Ponder, Marissa A. Weis, Andersen Chang, Taliah Muhammad, Saumil Patel, Zhiwei Ding, Dat Tran, Jiakun Fu, Casey M. Schneider-Mizell, Nuno Maçarico da Costa, R. Clay Reid, Forrest Collman, Katrin Franke, Alexander S. Ecker, Jacob Reimer, Xaq Pitkow, Fabian H. Sinz, and Andreas S. Tolias. Foundation model of neural activity predicts response to new stimulus types. 640(8058):470–477, 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-08829-y. URL https://www.nature.com/articles/s41586-025-08829-y.

Alex H. Williams, Tony Hyun Kim, Forea Wang, Saurabh Vyas, Stephen I. Ryu, Krishna V. Shenoy, Mark Schnitzer, Tamara G. Kolda, and Surya Ganguli. Unsupervised Discovery of Demixed, Low-Dimensional Neural Dynamics across Multiple Timescales through Tensor Component Analysis. 98(6):1099–1115.e8, 2018. ISSN 1097-4199. doi: 10.1016/j.neuron.2018.05.015.

Alex H. Williams, Erin Kunz, Simon Kornblith, and Scott W. Linderman. Generalized Shape Metrics on Neural Representations, 2022. URL http://arxiv.org/abs/2110.14739.

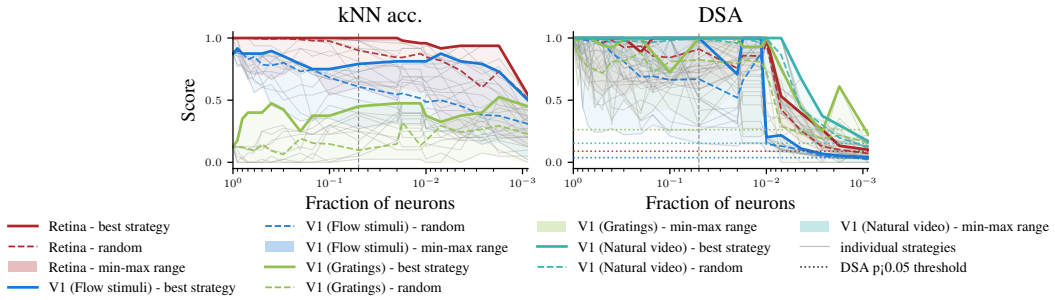

Figure 6: Fraction sweep for kNN accuracy and normalized DSA z-scores. Excluded from the main Figure 2 because k-NN is not a proper alignment metric and is unreliable with few points per class (undefined in the natural-video setting), and DSA is a normalized z-score rather than a bounded correlation-type score, so neither is directly comparable to the other six metrics on the same axis.

A Additional Results

We report k-NN accuracy and DSA separately from the six main-text metrics (Figure 2) rather than in the main figure, since both have a different interpretation from the other six. k-NN accuracy is not a proper alignment metric in the sense of the other six (it does not compare two representations against each other) and is more brittle in the low-samples-per-class regime; in the natural-video setting, where there are not multiple points per class, it is not meaningful at all. DSA scores, unlike the other metrics, are normalized z-scores relative to a temporal-shuffle null (Appendix B.10) rather than a bounded correlation-type score, so their absolute scale is not comparable to the other six metrics.

A.1 The Region-Sampling Effect Is Not a Coarse Excitatory/Inhibitory Split

One might expect Figure 3 to simply separate a highly selective, diversely tuned cell class (top row) from a broadly tuned, low-selectivity class (bottom row), analogous to excitatory vs. inhibitory neurons (Dyballa et al., 2024). The opposite holds: it is the *low*-decoding population (bottom row) that is more selective by standard tuning statistics, in every dataset.

Table 3: 5% high-/low-decoding subpopulations: OSI, lifetime sparseness, and response amplitude (hi/lo).

	OSI hi/lo	Lifetime sparseness hi/lo	Amplitude hi/lo
Retina	0.35 / 0.49	0.26 / 0.39	24.94 / 2.51
V1	0.35 / 0.69	0.31 / 0.35	10.53 / 1.40
VISp (G)	0.59 / 0.79	0.33 / 0.48	0.47 / 0.02
VISp (NV)	– / –	0.42 / 0.58	0.36 / 0.004

The only large, consistent separator between the two conditions is response amplitude (≈8–90×), not tuning selectivity or sparseness. We further checked whether restricting the region-sampling to a *single* functional group of the encoding manifold changes this picture:

Table 4: Retina, samples confined to a single functional group of the encoding manifold: decoding metrics still vary substantially while GW similarity stays comparatively low.

Group	k (% pop.)	RSA	CKA	Proc. R^2	GW
A	278 (24%)	0.979	0.964	0.863	0.227
B	195 (17%)	0.767	0.875	0.869	0.321
C	171 (15%)	0.594	0.685	0.526	0.199

Even samples confined to a single functional group vary substantially in decoding scores, while GW similarity remains comparatively low throughout – consistent with our claim that decoding alignment and encoding-manifold coverage are dissociable properties, not a proxy for a two-way cell-type split.

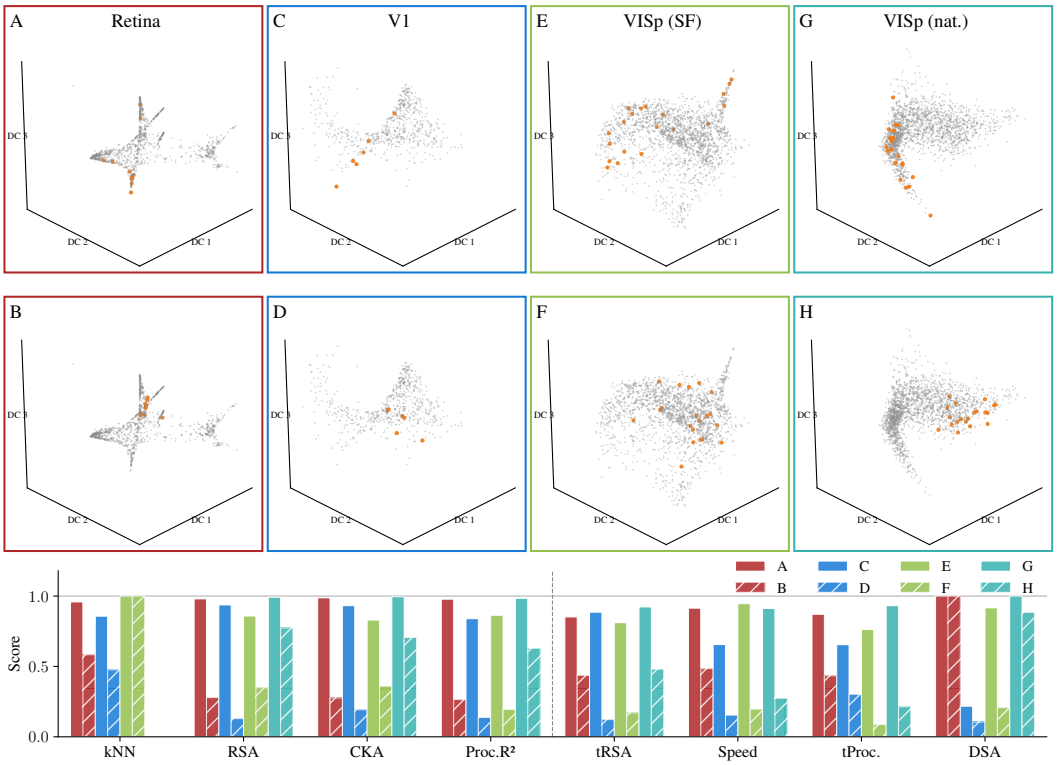

Figure 7: Size-matched encoding manifold region sampling for Retina, V1 and VISp (gratings and natural video). Samples drawn to maximize (top) or minimize (bottom) first principal component contribution of selected neurons, generating near best-/worst-case samples. These samples from distinct regions of the encoding manifold yield qualitatively different decoding profiles under the same eight metrics. Under high-PC selection, a small (1%) subregion sample can suffice to largely recover alignment metrics.

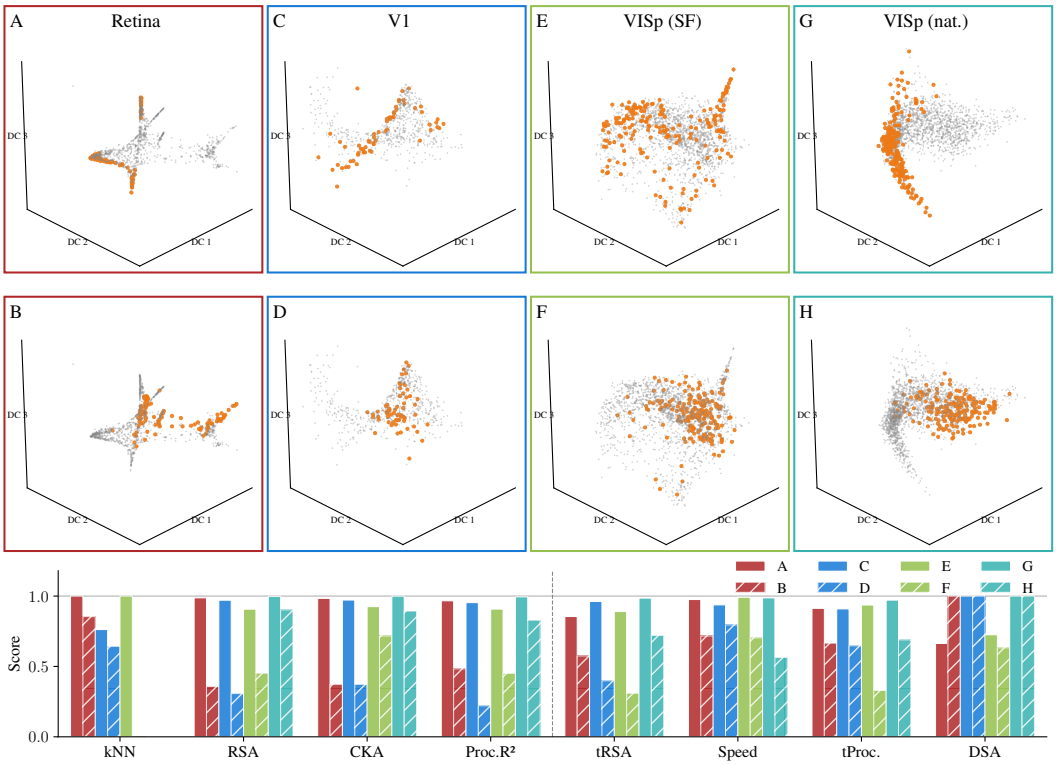

Figure 8: Size-matched encoding manifold region sampling for Retina, V1 and VISp (gratings and natural video). Samples drawn to maximize (top) or minimize (bottom) first principal component contribution of selected neurons, generating near best-/worst-case samples. These samples from distinct regions of the encoding manifold yield qualitatively different decoding profiles under the same eight metrics. Under high-PC selection, a small (10%) subregion sample can suffice to largely recover alignment metrics.

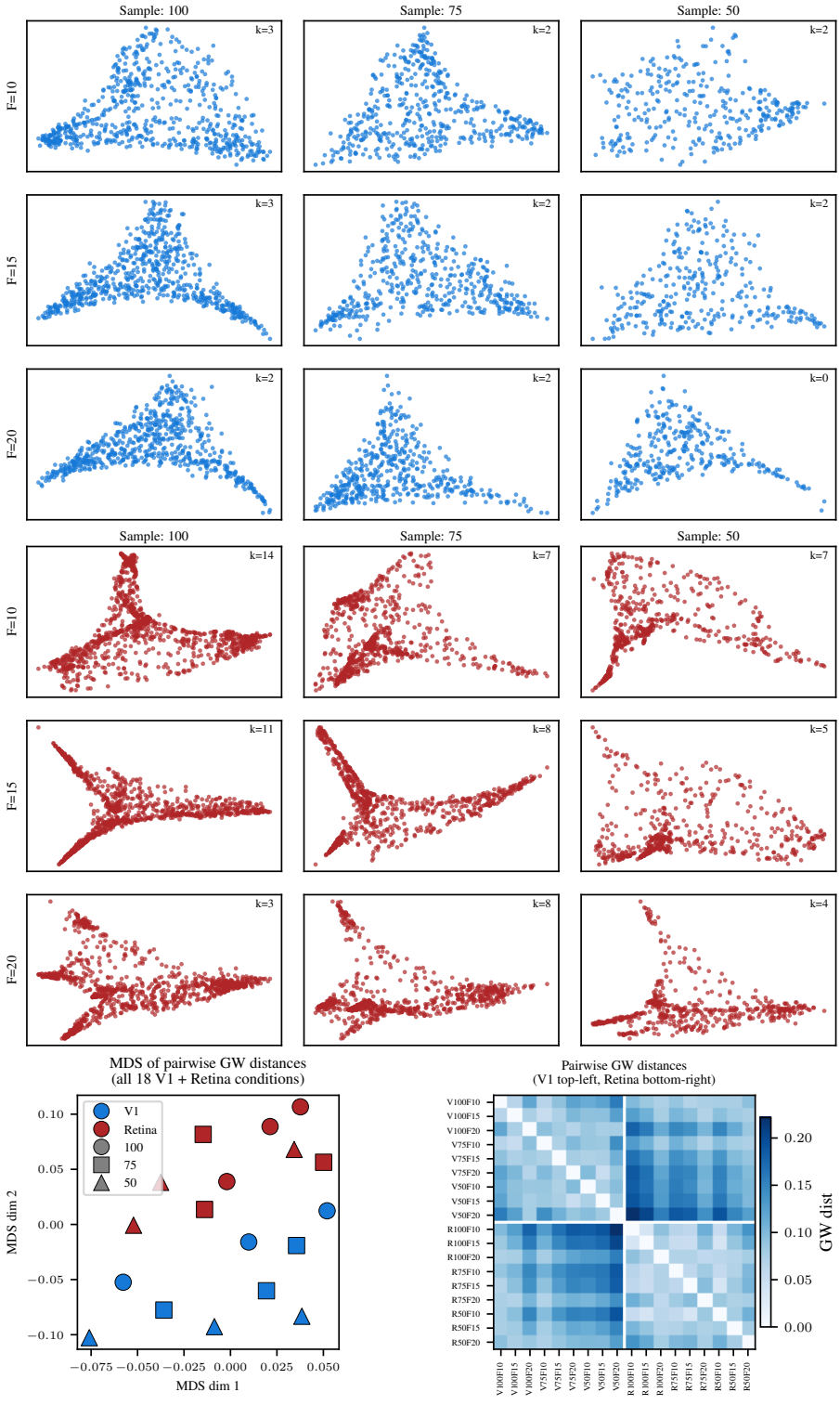

Figure 9: **Encoding manifold pipeline stability** across sample size (columns), tensor decomposition factors (rows) and datasets (Blue: V1, Red: Retina). Encoding topology is stable, showing continuous manifolds in V1 and clustered ones in retina. GW distance groups the conditions of each dataset together (bottom).

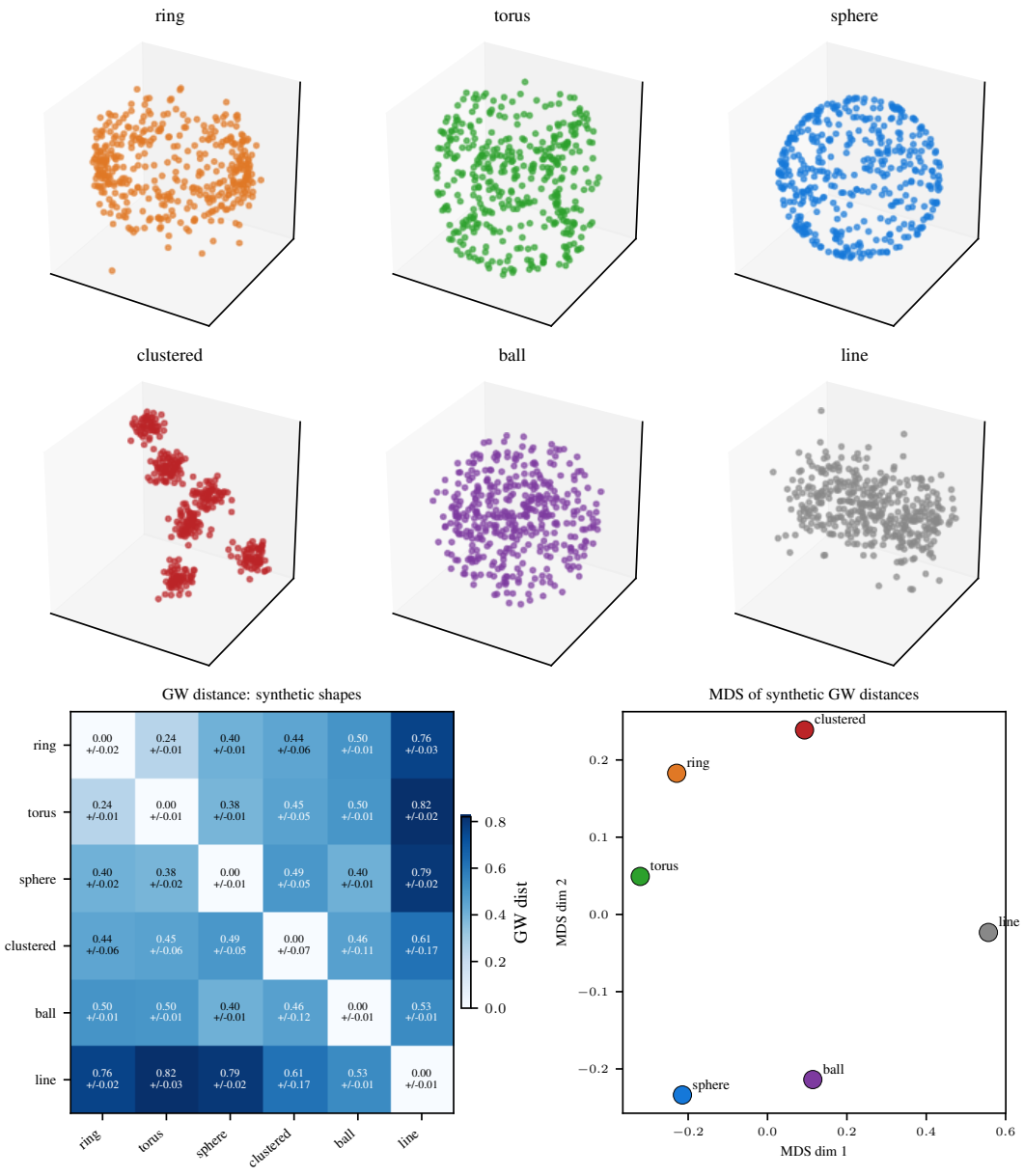

Figure 10: **GW intuition** based on simple synthetic datasets. The ring and torus being topologically related are grouped together. Similarly, the sphere and ball are close in GW space. The line is furthest from all other datasets, not allowing for circular matching of points.

Table 5: MNIST λ-sweep: encoding topology is progressively clustered while decoding metrics remain near-ceiling. All scores are relative to the baseline ($\lambda = 0$). GW Sim (IAN) measures normalized encoding manifold similarity via Gromov–Wasserstein distance. Single model seed, each λ compared with the same-seed baseline (hence baseline $= 1$); Table 2 instead uses 20 seeds per condition, and its baseline column compares different baseline seeds (GW 0.958).

λ	Acc	RSA	CKA	Proc. R^2	GW Sim (IAN)
0 (baseline)	0.985	1.000	1.000	1.000	1.000
0.1	0.986	0.997	0.998	0.997	0.970
0.5	0.986	0.997	0.998	0.998	0.941
1	0.985	0.996	0.998	0.997	0.943
5	0.985	0.982	0.990	0.989	0.706
10	0.984	0.968	0.982	0.981	0.466
50	0.983	0.897	0.936	0.925	0.284

Table 6: Gaussian blur robustness: test accuracy at each blur level for baseline (λ=0) vs. clustered (λ=10), n=20 seeds (mean $\pm$ SEM, Wilcoxon signed-rank, paired by seed; ns = $p \geq 0.05$).

σ_{blur}	Baseline	Clustered (λ=10)	p
0	0.985 ± 0.000	0.983 ± 0.000	<0.001 ***
2	0.878 ± 0.007	0.884 ± 0.009	0.409 ns
3	0.639 ± 0.014	0.622 ± 0.016	0.294 ns
4	0.446 ± 0.011	0.401 ± 0.015	0.015 *
5	0.304 ± 0.013	0.259 ± 0.014	0.007 **
6	0.220 ± 0.012	0.193 ± 0.013	0.083 ns

A.2 FPS Subpopulation Showcases

Figures 11–25 show encoding manifold, decoding manifold, and stimulus trajectory visualizations for FPS continuous and clustered subpopulations across all biological systems and machine learning models, together with the full per-dataset decoding metric bar charts. Table 7 reports all eight metrics numerically across all 17 datasets.

A.2.1 Biological Systems

A.2.2 Machine Learning Models

A.3 Comparison with the Soft-Matching Distance

We compare the MNIST models of Section 4.4 with the Soft-Matching Distance (SMD) (Khosla & Williams, 2024), which matches single-neuron tuning vectors across two networks. Each of the 800 flatten-layer units is represented by its responses to the same 500 test images (50 per digit; the images of Table 2), z-scored across images. SMD is the square root of the exact optimal-transport cost with uniform marginals (`ot.emd`), with the squared Euclidean distance divided by the number of images as cost. Hence $\text{SMD} \in [0, 2]$, and $\text{SMD} = \sqrt{2}$ corresponds to uncorrelated matched units. Units with zero variance over the 500 images (0–25 of 800 per model) are excluded.

Models trained with the same seed share their initial weights. For a neuron-level measure, a baseline and a clustered model with the same seed are therefore not an independent pair. Units with the same index in Base_i and Clust_i have a mean tuning correlation of 0.70, compared with 0.16 for models of different seeds. We therefore compare Base_i vs. Clust_{i+1} with Base_i vs. Base_{i+1} (indices mod 20). In both pairs, the second model has the same initialization and differs only in its training objective (Wilcoxon signed-rank, paired by i, n=20, Holm-corrected; Table 8).

SMD separates the conditions: in all 20 pairs, a baseline and a clustered model are further apart than two baseline models (0.772 vs. 0.711). This holds without z-scoring (Table 8) and when dead units are kept rather than dropped (0.771 vs. 0.710). With seeds controlled, however, RSA, CKA, and Procrustes R^2 also decrease consistently (in 20, 20, and 17 of 20 pairs), although their absolute values stay above 0.91. The null differences for these metrics in Table 2 arise because same-seed pairs share

Table 7: Subpopulation metric scores across 17 datasets. Top rows (continuous): FPS 50 seeds × 1 NN; Bottom rows (Clustered): 10 seeds × 9 NN. Values show mean ± std over 5 seeds.

Dataset	kNN	RSA	CKA	Proc.R^2	tRSA	Spd.	tProc	DSA
VISp G	0.80±.40	0.88±.03	0.92±.04	0.90±.02	0.76±.02	0.94±.01	0.69±.01	0.88±.10
Cluster	0.20±.40	0.88±.02	0.93±.01	0.92±.01	0.75±.02	0.94±.01	0.68±.03	0.74±.20
VISrl G	1.00	0.78±.03	0.89±.03	0.87±.02	0.65±.02	0.94±.01	0.62±.01	0.42±.20
Cluster	1.00	0.77±.03	0.91±.03	0.88±.01	0.65±.02	0.93±.01	0.61±.01	0.33±.07
VISam G	0.87±.27	0.79±.03	0.87±.01	0.85±.01	0.69±.02	0.96±.01	0.72±.03	0.70±.08
Cluster	1.00	0.83±.02	0.90±.02	0.88±.01	0.73±.02	0.96±.01	0.72±.01	0.79±.19
VISpm G	0.20±.24	0.81±.03	0.90±.01	0.89±.01	0.74±.03	0.94±.01	0.71±.03	0.49±.16
Cluster	0.00	0.87±.01	0.91±.01	0.90±.01	0.79±.01	0.94±.01	0.70±.03	0.41±.10
VISl G	0.40±.49	0.73±.06	0.89±.03	0.89±.02	0.66±.05	0.96±.01	0.71±.04	0.71±.18
Cluster	0.00	0.79±.03	0.90±.01	0.88±.01	0.72±.02	0.96±.01	0.73±.02	0.63±.18
V1 NV	–	0.99±.01	0.99±.01	0.98±.01	0.94±.01	0.86±.01	0.90±.01	1.00
Cluster	–	0.98±.01	0.98±.01	0.97±.01	0.94±.01	0.87±.01	0.90±.01	1.00
VISam NV	–	0.99±.00	0.99±.00	0.99±.00	0.96±.00	0.84±.01	0.90±.01	0.96±.04
Cluster	–	0.99±.00	1.00±.00	0.99±.00	0.97±.00	0.86±.01	0.90±.01	1.00
VISrl NV	–	0.99±.01	0.99±.01	0.99±.00	0.96±.00	0.84±.01	0.88±.01	1.00
Cluster	–	0.99±.00	0.99±.00	0.99±.00	0.96±.00	0.84±.02	0.88±.01	1.00
VISpm NV	–	0.99±.00	0.99±.00	0.99±.00	0.96±.01	0.85±.01	0.90±.01	0.96±.05
Cluster	–	0.99±.01	0.99±.01	0.99±.01	0.96±.01	0.84±.03	0.89±.02	0.94±.08
VISl NV	–	0.98±.01	0.97±.02	0.96±.01	0.94±.01	0.84±.01	0.89±.01	0.74±.04
Cluster	–	0.97±.03	0.96±.04	0.95±.02	0.94±.01	0.84±.01	0.88±.01	0.70±.05
V1 F	0.87±.05	0.91±.02	0.92±.02	0.86±.02	0.84±.03	0.90±.02	0.87±.01	0.88±.14
Cluster	0.88±.03	0.93±.01	0.93±.01	0.88±.01	0.86±.01	0.93±.01	0.88±.00	0.87±.17
Retina	0.98±.02	0.89±.03	0.97±.01	0.96±.01	0.87±.01	0.94±.00	0.93±.01	0.81±.11
Cluster	0.99±.01	0.95±.03	0.98±.02	0.97±.01	0.91±.01	0.93±.01	0.92±.00	0.82±.10
FNN	0.92±.06	0.88±.04	0.94±.02	0.89±.01	0.80±.03	0.97±.00	0.87±.01	0.63±.24
Cluster	0.95±.06	0.75±.09	0.85±.08	0.83±.04	0.71±.06	0.97±.00	0.89±.01	0.60±.29
Flyvision	0.95±.04	0.96±.00	0.94±.00	0.85±.01	0.94±.00	0.82±.01	0.89±.00	1.00
Cluster	0.98±.01	0.94±.00	0.88±.02	0.82±.01	0.92±.01	0.80±.01	0.87±.01	0.98±.03
R2+1D L4	1.00±.01	0.82±.04	0.93±.01	0.93±.01	0.84±.03	0.73±.04	0.92±.01	0.69±.06
Cluster	1.00	0.76±.06	0.89±.02	0.88±.01	0.78±.04	0.69±.02	0.90±.01	0.68±.06
ViT	0.90±.07	0.94±.02	0.85±.08	0.82±.04	0.73±.08	0.47±.07	0.76±.01	1.00
Clustered	0.87±.04	0.98±.00	0.96±.01	0.87±.01	0.87±.02	0.60±.05	0.80±.01	1.00
Raptor	0.84±.02	0.65±.01	0.83±.00	0.70±.00	0.56±.01	1.00±.00	0.99±.00	0.37±.03
Clustered	0.86±.01	0.69±.00	0.84±.00	0.70±.00	0.57±.01	1.00±.00	0.98±.00	0.30±.03

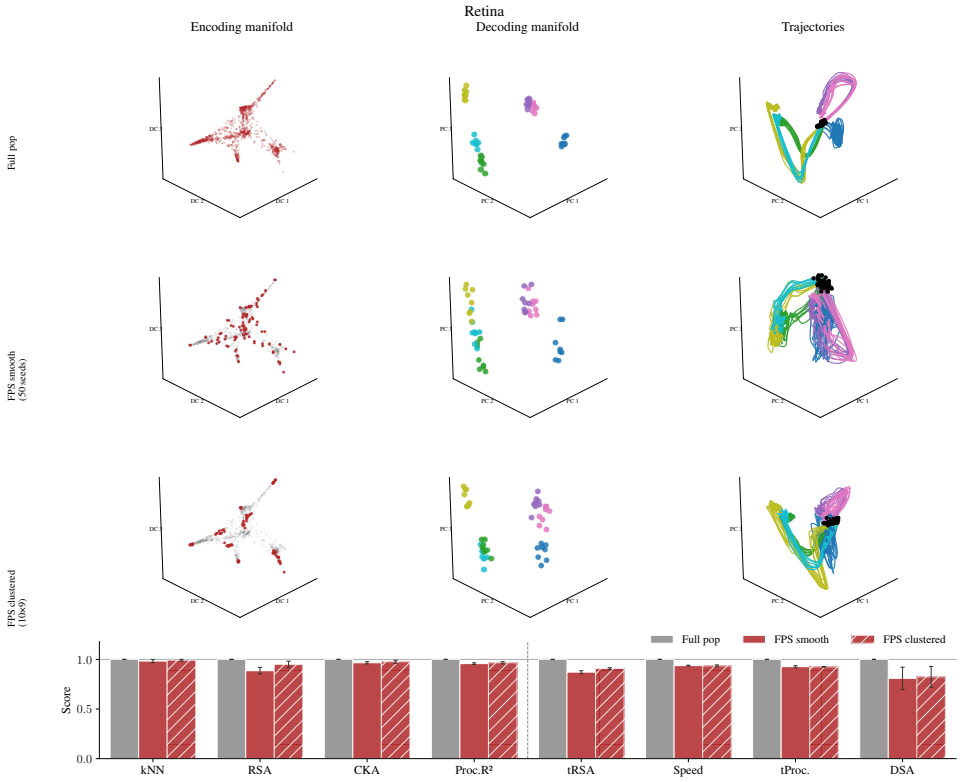

Figure 11: **Retina — FPS subpopulation showcase.** Each row shows the 3-D encoding manifold (left), 3-D decoding manifold (center), and stimulus trajectories (right) for the full population (top), FPS continuous subpopulation (n=50 seeds, m=1 neighbor, ≈100 neurons; middle), and FPS clustered subpopulation (n=10 seeds, m=9 neighbors, ≈100 neurons; bottom). The bottom bar chart reports all eight decoding metrics relative to the full population (mean ± std over five random seeds {42–46}).

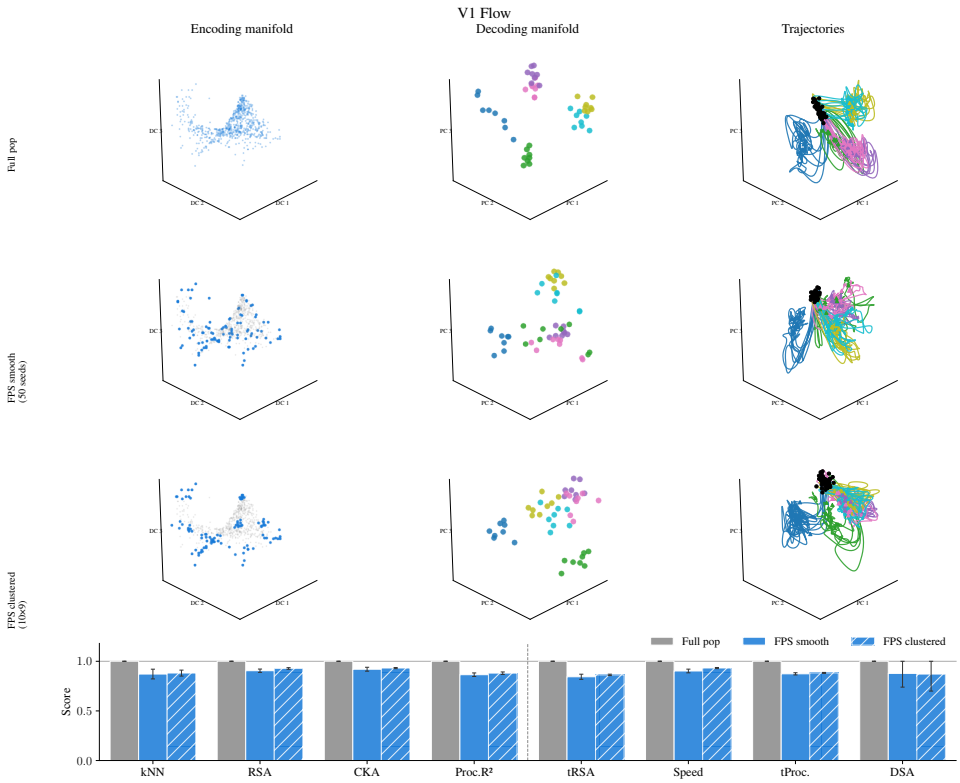

Figure 12: **V1 — FPS subpopulation showcase.** Layout as in Figure 11. FPS continuous: n=50, m=1; FPS clustered: n=10, m=9. Both yield ≈100 neurons.

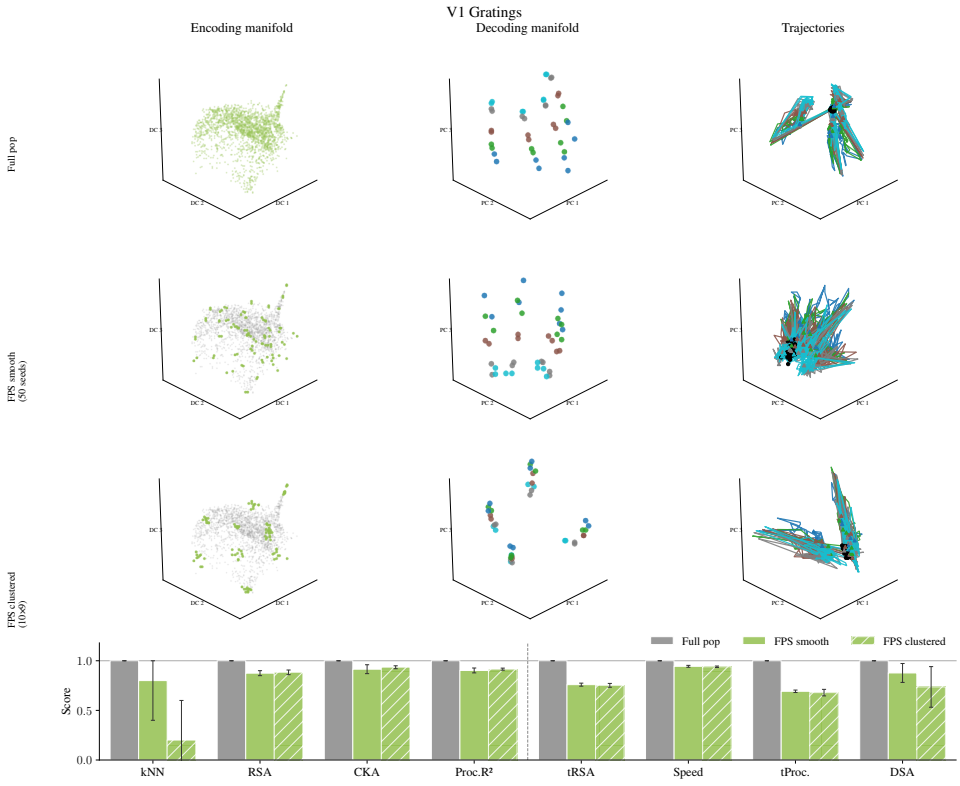

Figure 13: **Allen VISp (drifting gratings) — FPS subpopulation showcase.** Layout as in Figure 11. FPS continuous: n=50, m=1; FPS clustered: n=10, m=9.

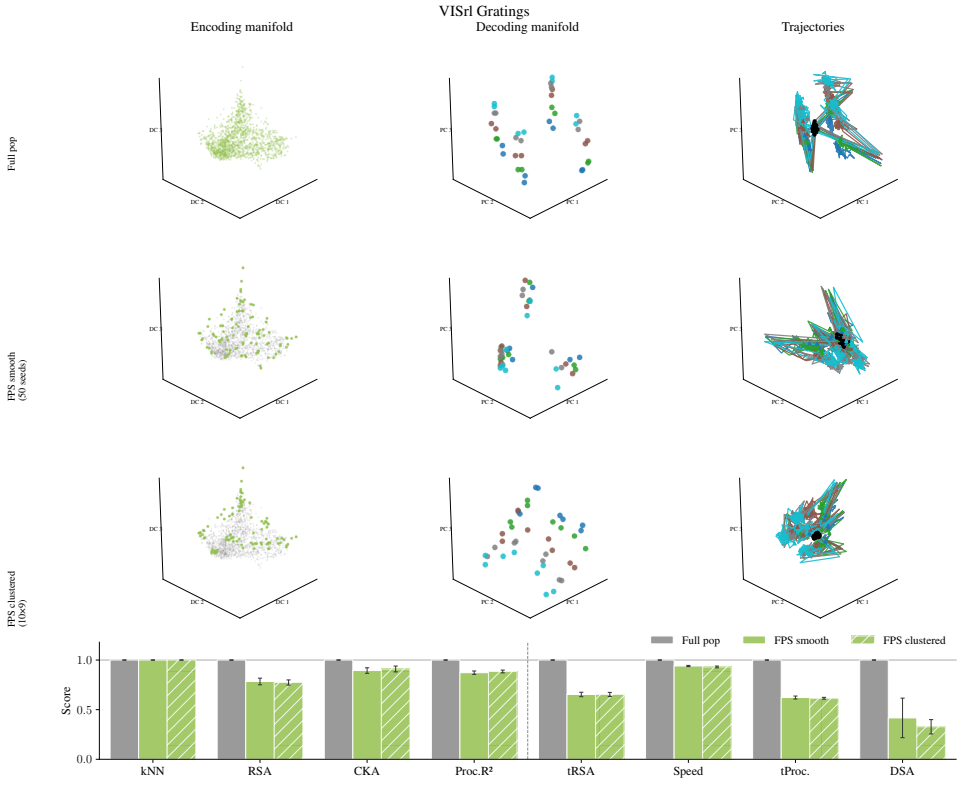

Figure 14: **Allen VISrl (drifting gratings) — FPS subpopulation showcase.** Layout as in Figure 11.

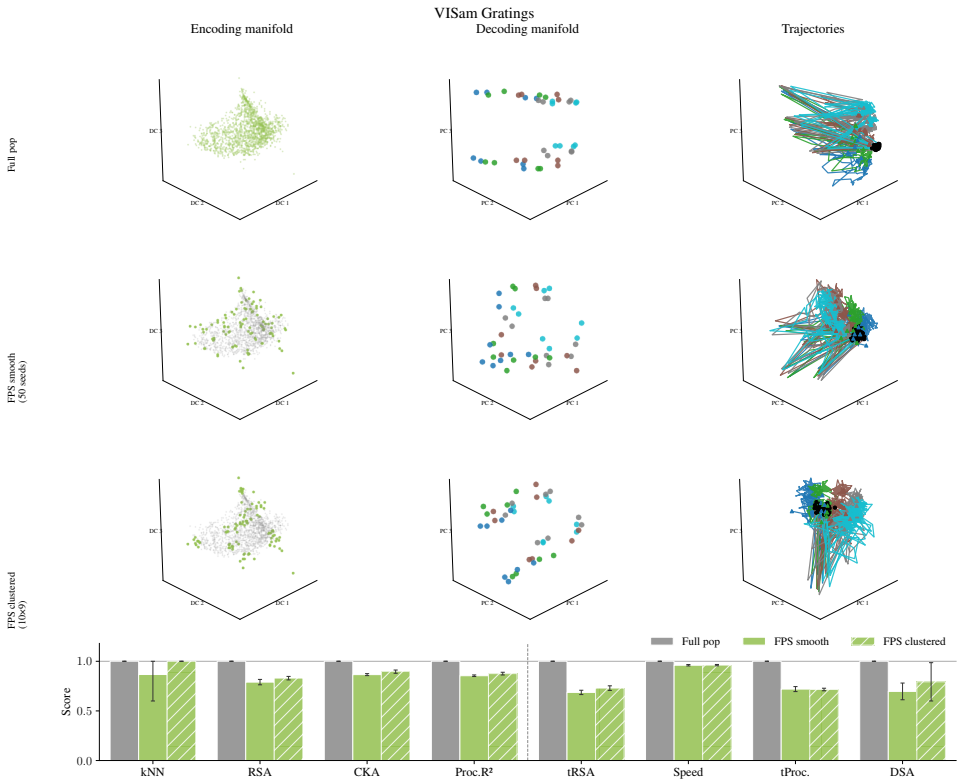

Figure 15: **Allen VISam (drifting gratings) — FPS subpopulation showcase.** Layout as in Figure 11.

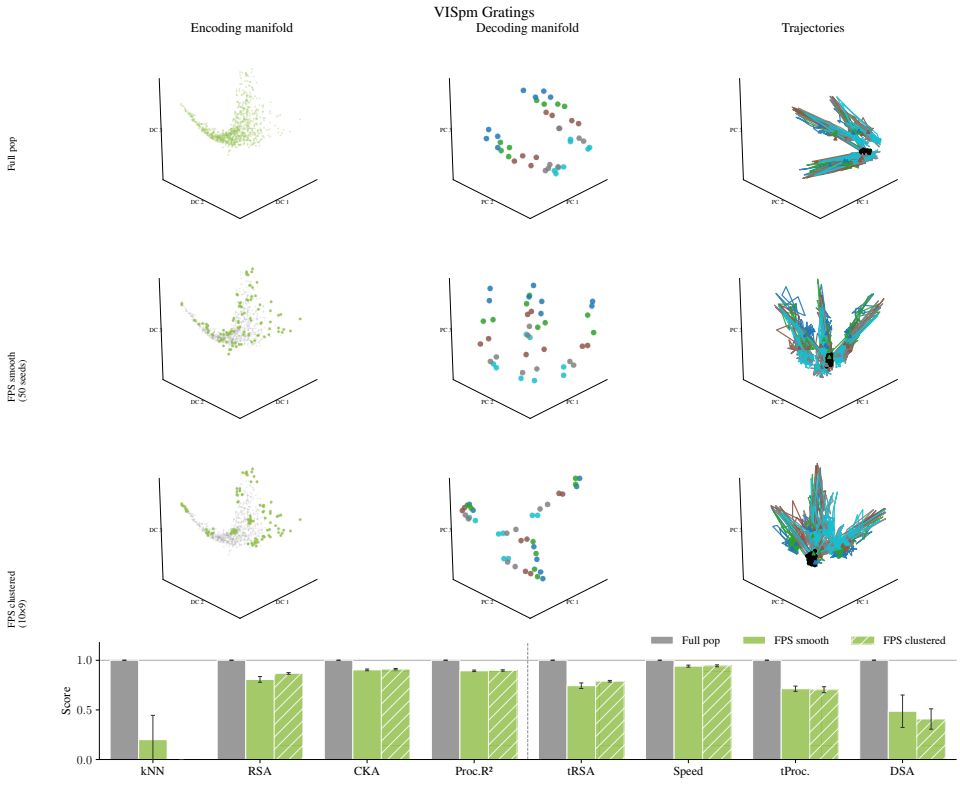

Figure 16: **Allen VISpm (drifting gratings) — FPS subpopulation showcase.** Layout as in Figure 11.

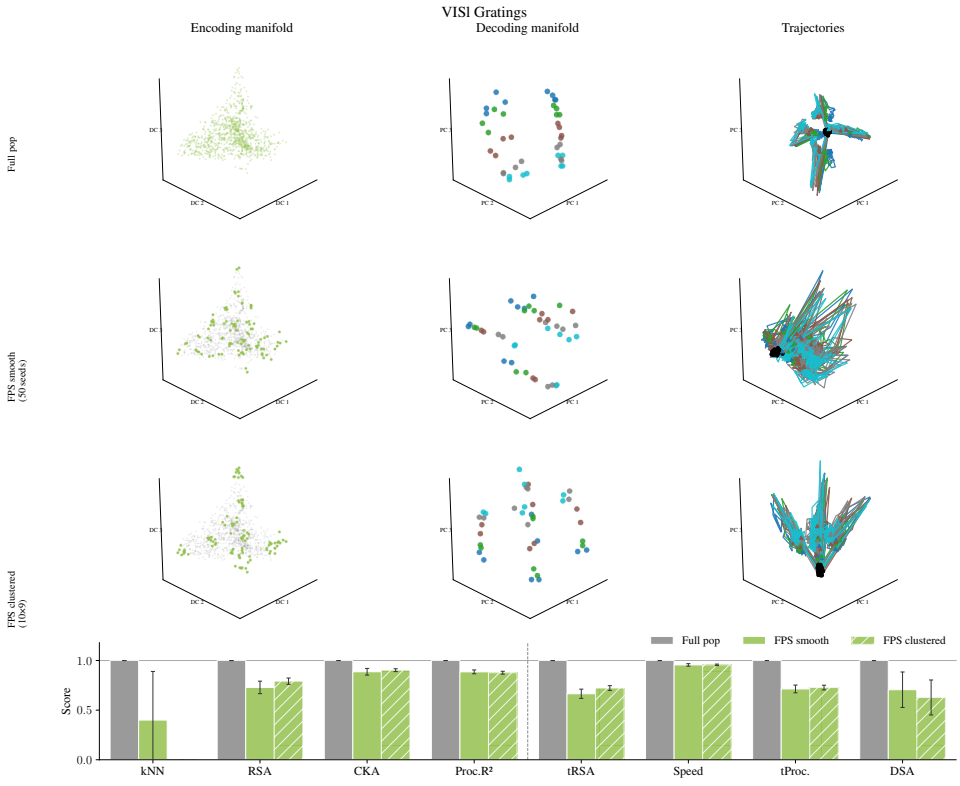

Figure 17: **Allen VISl (drifting gratings) — FPS subpopulation showcase.** Layout as in Figure 11.

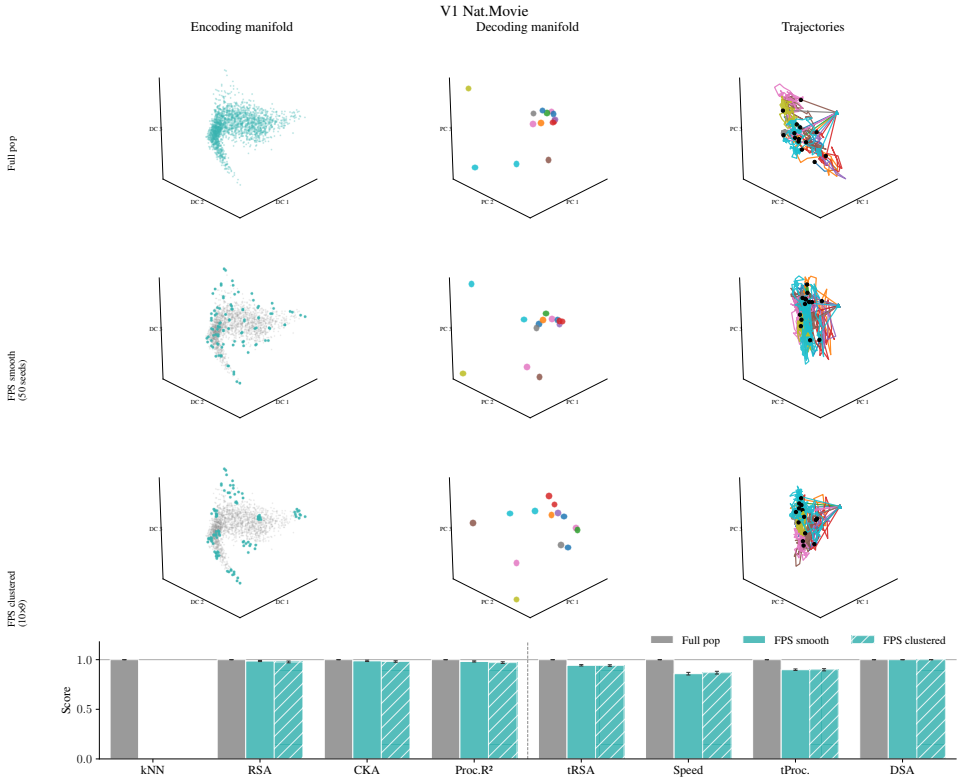

Figure 18: **Allen VISp (natural movie, 156 scenes) — FPS subpopulation showcase.** Layout as in Figure 11. Because the natural movie stimulus has a single direction (D=1), k-NN accuracy is not computed (marked "–" in the bar chart).

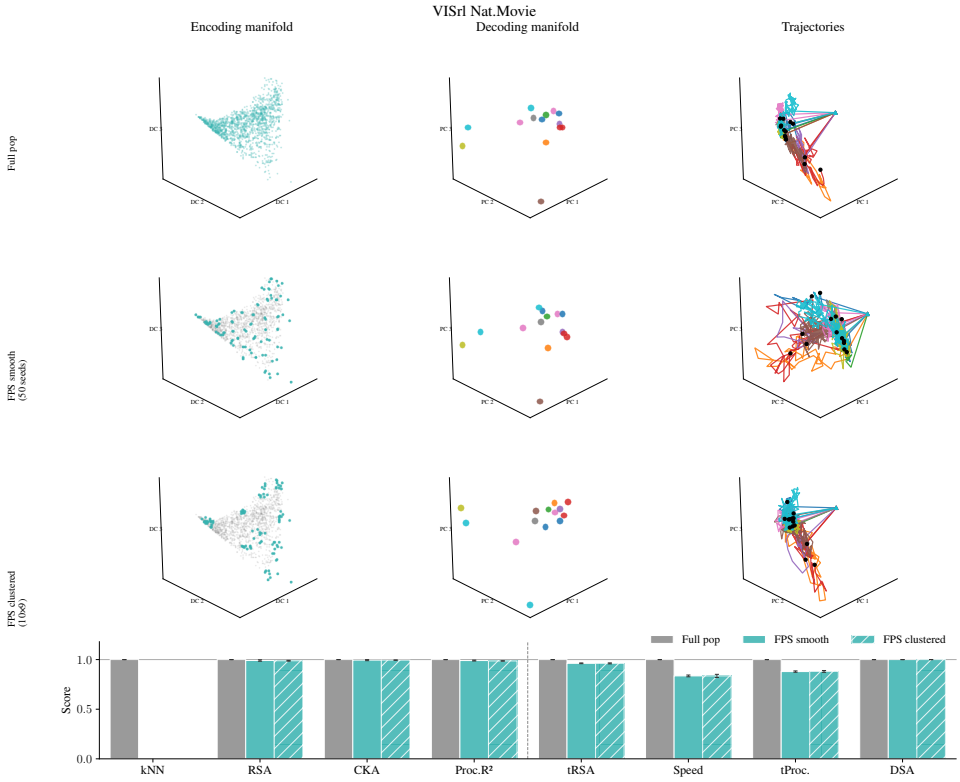

Figure 19: **Allen VISrl (natural movie) — FPS subpopulation showcase.** Layout as in Figure 18.

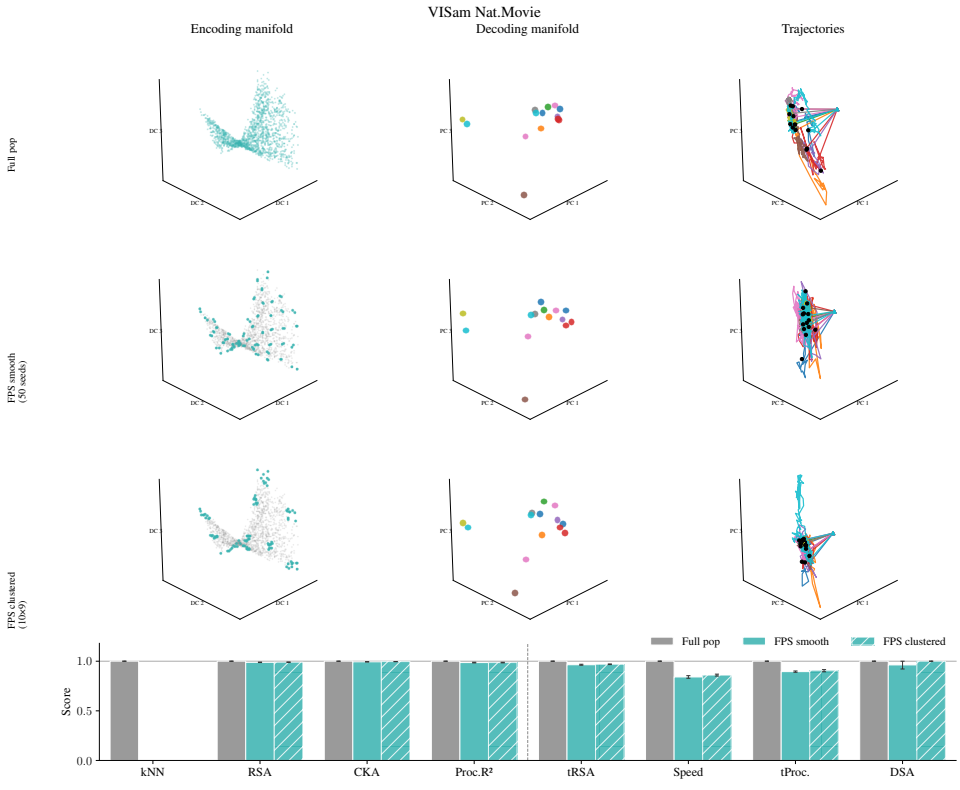

Figure 20: **Allen VISam (natural movie) — FPS subpopulation showcase.** Layout as in Figure 18.

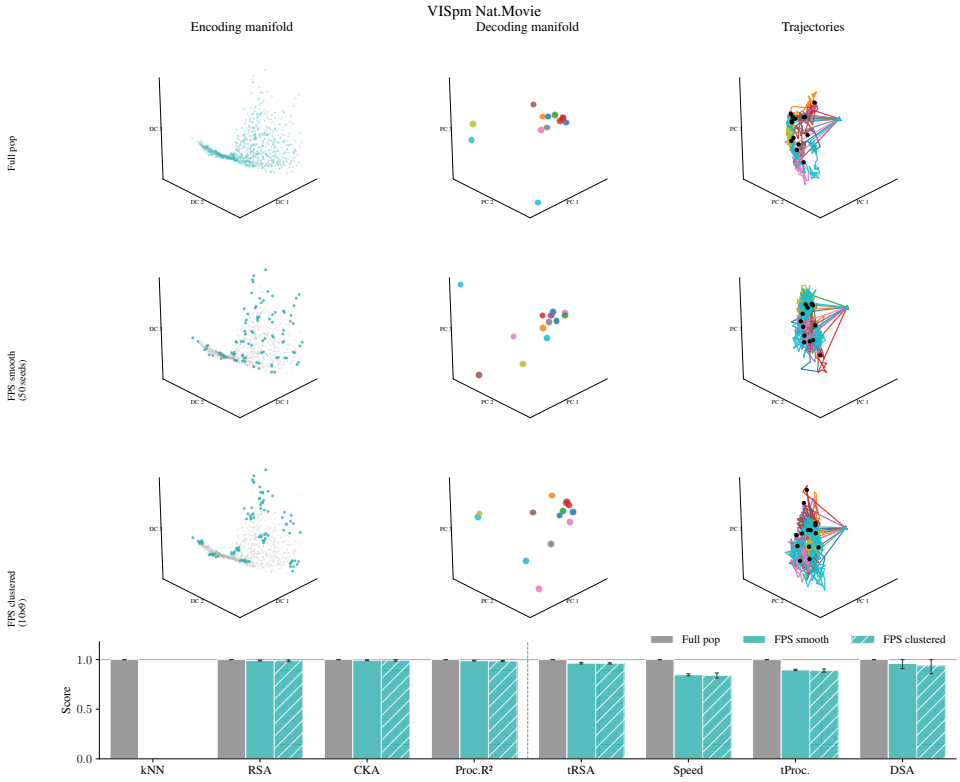

Figure 21: **Allen VISpm (natural movie) — FPS subpopulation showcase.** Layout as in Figure 18.

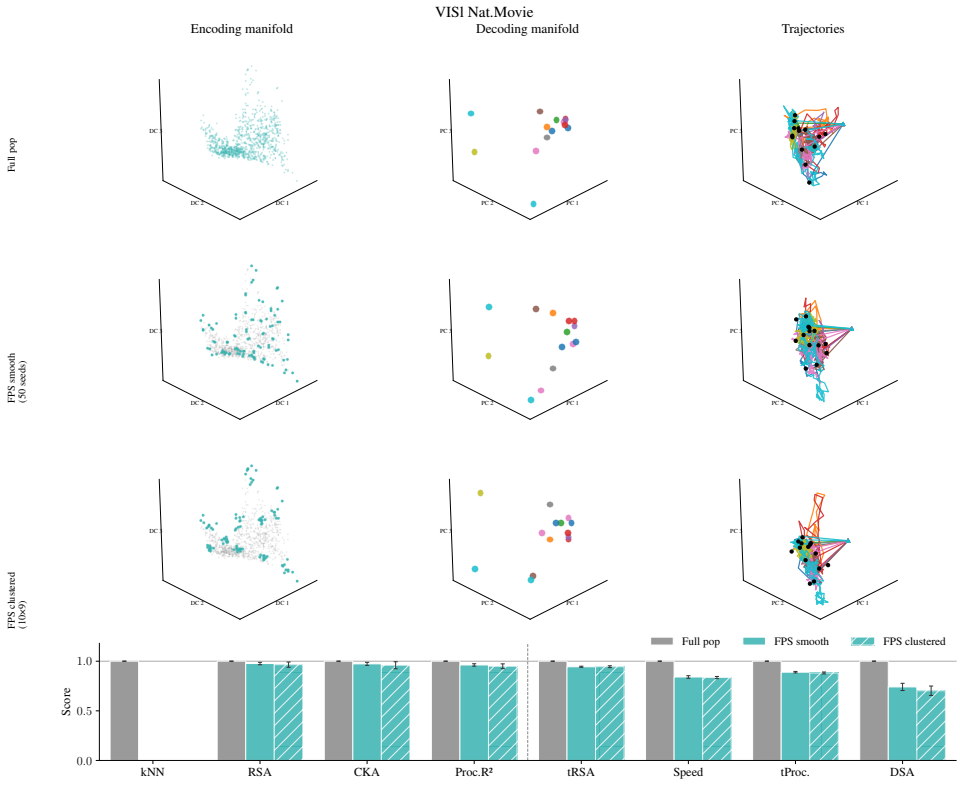

Figure 22: **Allen VISl (natural movie) — FPS subpopulation showcase.** Layout as in Figure 18.

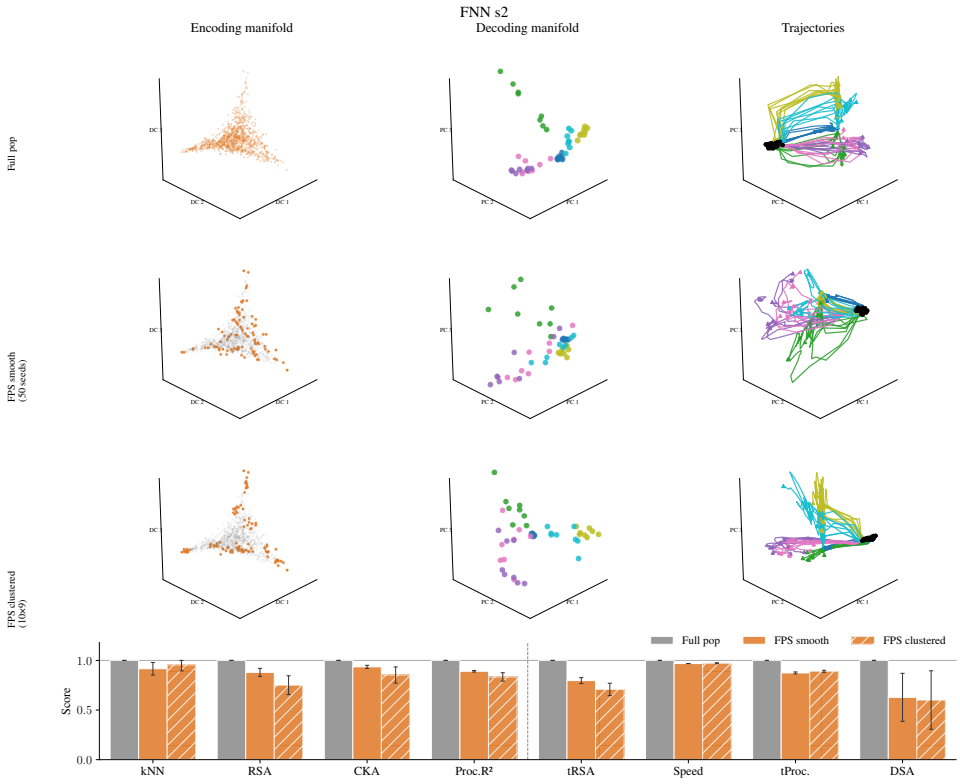

Figure 23: **FNN output layer — FPS subpopulation showcase.** Layout as in Figure 11. 2000 sampled neurons from the FNN recurrent output layer, probed with the 6-class × 8-direction optical-flow stimulus ensemble.

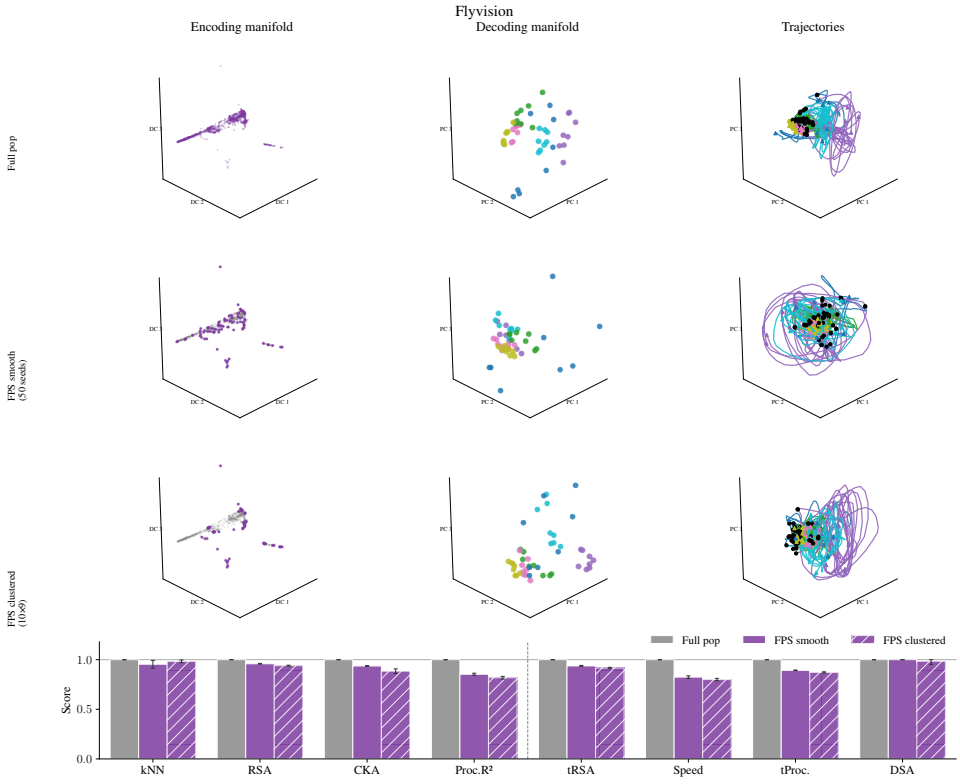

Figure 24: **Flyvision (T, Tm cells) — FPS subpopulation showcase.** Layout as in Figure 11. 1700 neurons of the T and Tm cell types from a connectome-constrained fly visual system model (Lappalainen et al., 2024).

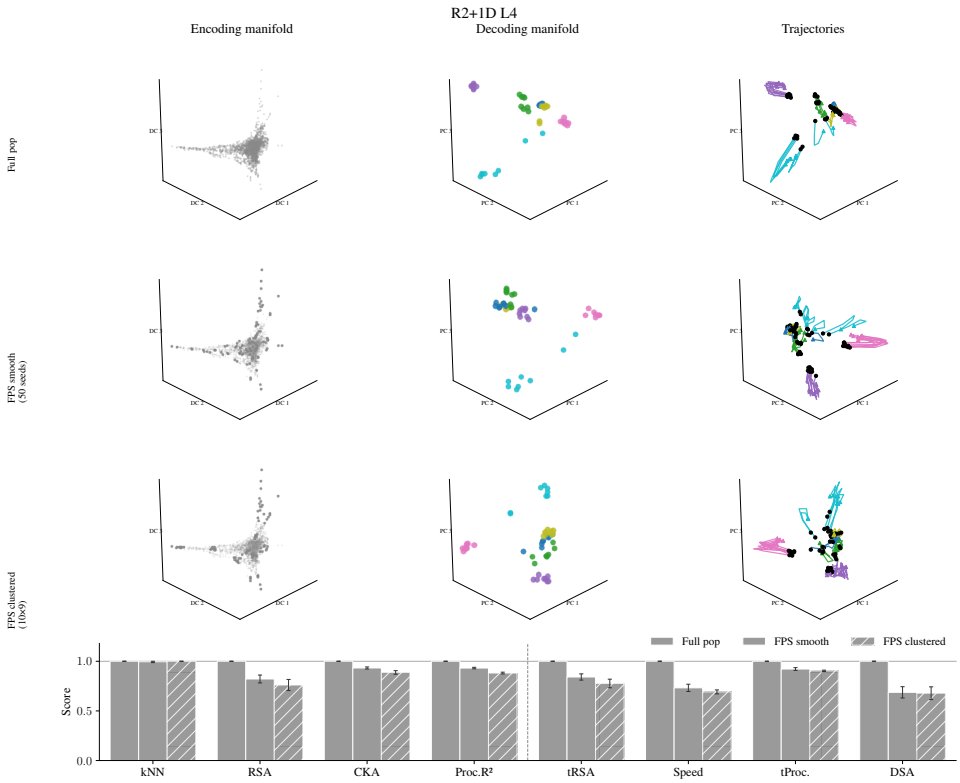

Figure 25: **R(2+1)D layer 4 — FPS subpopulation showcase.** Layout as in Figure 11. 1960 activations from layer 4 of a pretrained spatiotemporal video CNN (Tran et al., 2018).

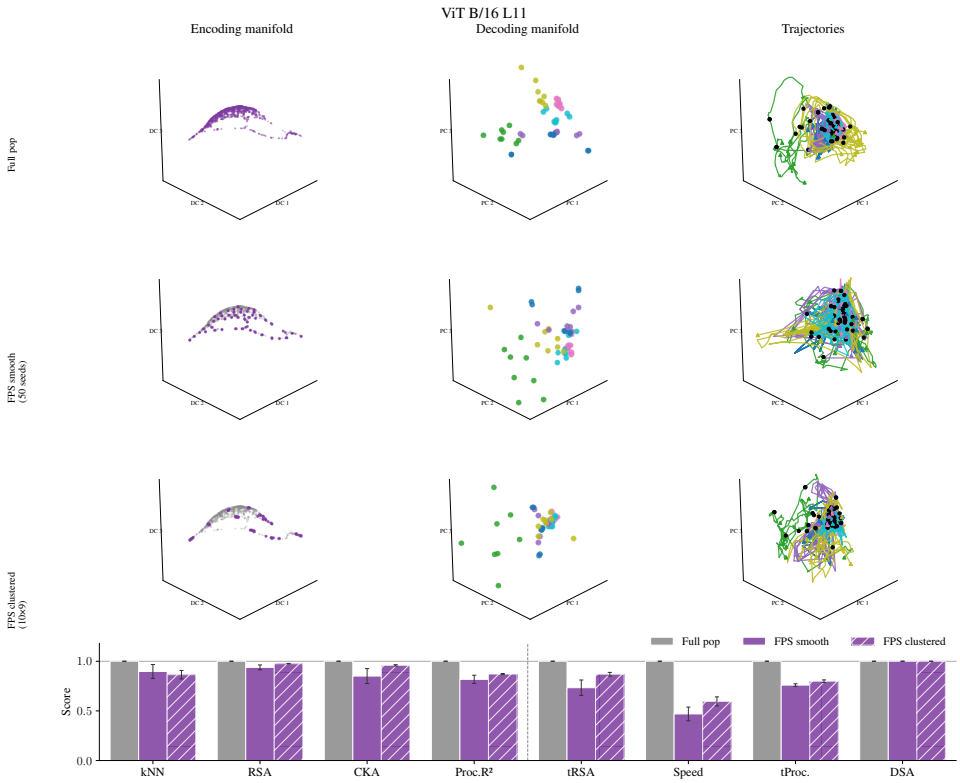

Figure 26: **ViT Layer 11 — FPS subpopulation showcase.** Layout as in Figure 11. Activations from a pretrained vision transformer (Dosovitskiy et al., 2021).

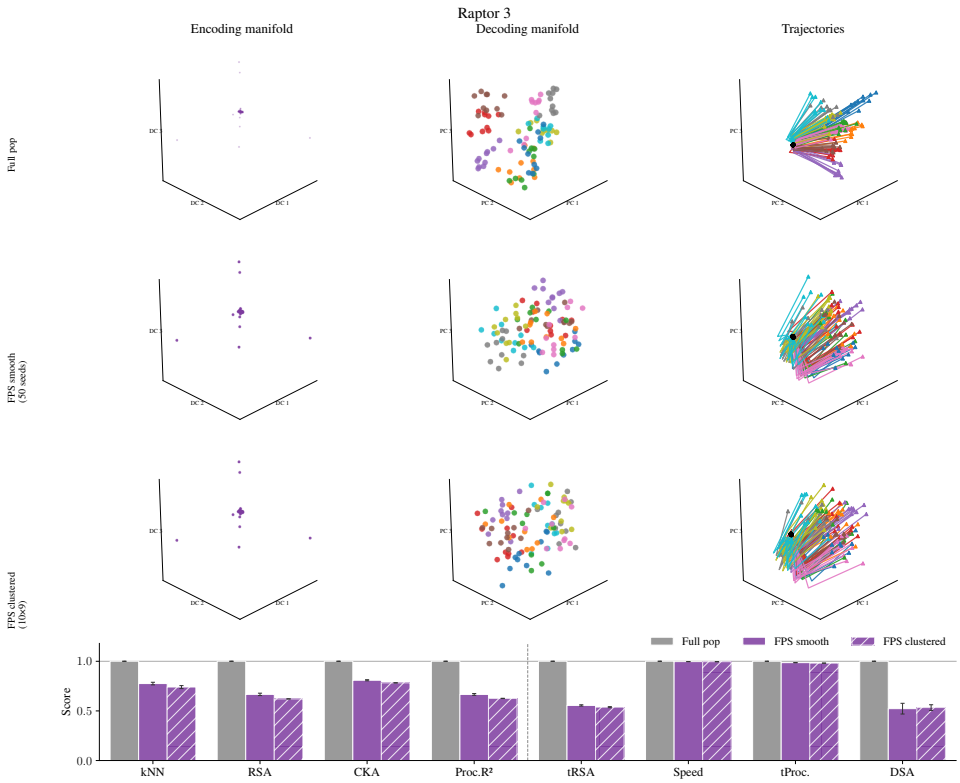

Figure 27: **Raptor — FPS subpopulation showcase.** Layout as in Figure 11. Activations from a pretrained recurrent vision transformer (Jacobs et al., 2026).

Table 8: Soft-Matching Distance (SMD) and decoding metrics for the MNIST models (mean ± SEM over $i = 0, \ldots, 19$). p: Base_i vs. Clust_{i+1} against Base_i vs. Base_{i+1} (Wilcoxon signed-rank, Holm-corrected). Seeds: number of i for which the cross-condition pair is less similar. ‡GW is available only for same-seed pairs (Table 2); its test uses Base_i vs. Clust_i. †Uncorrected.

Metric	Base_i vs. Base_{i+1}	Base_i vs. Clust_i	Base_i vs. Clust_{i+1}	p	Seeds
RSA ↑	0.955 ± 0.004	0.962 ± 0.003	0.925 ± 0.006	9.5×10^{-6}	20/20
CKA ↑	0.975 ± 0.002	0.976 ± 0.002	0.956 ± 0.003	9.5×10^{-6}	20/20
Procrustes R^2 ↑	0.948 ± 0.008	0.943 ± 0.010	0.917 ± 0.010	0.005	17/20
SMD ↓	0.711 ± 0.005	0.682 ± 0.005	0.772 ± 0.005	9.5×10^{-6}	20/20
GW Sim ↑	0.957 ± 0.002	0.502 ± 0.022	–	$9.5\times10^{-6\ddagger}$	20/20
SMD, raw responses ↓	1.100 ± 0.012	1.034 ± 0.011	1.198 ± 0.014	$1.9\times10^{-6\dagger}$	20/20
Same-index unit corr.	0.164 ± 0.011	0.699 ± 0.007	0.163 ± 0.011	$0.87^{\dagger}$	–

their initialization, which raises their similarity to the level of two baseline models. We express each shift in units of the standard deviation (SD) of the Base_i vs. Base_{i+1} distribution. The reorganization moves RSA by 1.7 SDs, CKA by 2.5, Procrustes R^2 by 0.9, and SMD by 2.5. GW similarity falls from 0.957 to 0.502 (42 SDs), and it does so for same-seed pairs, where shared initialization raised the similarity of every other metric. Thus, the neuron-level SMD detects the reorganization, as does GW, and the reorganization also leaves a small but consistent trace in rotation-invariant metrics. GW differs in the size of the effect relative to seed-to-seed variability, in that it needs no shared stimulus frame, and in that it measures a different but related property (Section 2).

B METHODS

This section provides full methodological details for all experiments, datasets, and pipelines used in this paper.

B.1 DATA

B.1.1 BIOLOGICAL DATA: RETINA AND V1

Retinal and V1 neural recordings are taken from Dyballa et al. (2024). The stimulus ensemble consists of 88 unique sequences of drifting square-wave gratings and optical flows moving in 8 directions, grouped into 6 base-stimulus classes at two spatial frequencies (medium and high). For each neuron, the spatial frequency eliciting the larger population response is selected. The response tensor has shape $(N \times S \times K \times T)$, where N is the number of neurons (Retina: $N = 1{,}139$; V1: $N = 630$, after outlier removal in the encoding pipeline), S is the number of stimuli (6 after spatial-frequency selection), $K = 8$ is the number of movement directions, and T is the number of time bins ($T = 135$ for biological data at ~ 7.5 ms resolution).

B.1.2 ALLEN BRAIN OBSERVATORY

We use Neuropixels electrophysiology data from the Allen Brain Observatory Visual Coding dataset (de Vries et al., 2020), accessed via the AllenSDK `EcephysProjectCache`. We select all 32 sessions of the `brain_observatory_1.1` type, which include full 8-direction $\times$ 5-temporal-frequency drifting grating sweeps (~ 15 repeats per condition) and natural movie stimuli.

Drifting gratings. For each of the five cortical visual areas (VISp, VISl, VISam, VISpm, VISrl), units are pooled across sessions after quality filtering (ISI violations < 0.5, presence ratio > 0.9, amplitude > 50 μV). Trial-averaged PSTHs are computed in 50 ms bins over a 2-second window, yielding tensors of shape $(N_{\text{area}} \times 5 \times 8 \times 40)$, where the 5 stimuli correspond to temporal frequencies of 1, 2, 4, 8, and 15 Hz (spatial frequency fixed at 0.04 cpd, contrast 0.8).

Natural movie. Natural movie one (30 s clip, ~ 900 frames at 30 fps) is segmented into 156 non-overlapping scenes of ~ 6 frames each. Trial-averaged responses are computed per scene in the same 50 ms bins, yielding tensors of shape $(N_{\text{area}} \times 156 \times 1 \times T_{\text{scene}})$.

B.1.3 MACHINE LEARNING MODELS

We additionally analyze subpopulations in the following models:

- **FNN** (Foundation Neural Network) (Wang et al., 2025): We sample 2000 neurons from the recurrent hidden state and from the output layer of the pretrained model, probed with the same 6-class $\times$ 8-direction stimulus ensemble as the biological data. Tensor shape: $(2000 \times 6 \times 8 \times 37)$.
- **Flyvision** (Lappalainen et al., 2024): T and Tm cell types from a connectome-constrained fly visual system model (1700 neurons, same stimulus protocol).
- **R(2+1)D** (Tran et al., 2018): Layer 4 activations of a pretrained spatiotemporal video network (1960 neurons).
- **ViT-B/16** (Dosovitskiy et al., 2021): A standard Vision Transformer with no recurrent structure. The same 37-frame pixel-roll sequences are used; frames are fed individually to the model and activations from a single encoder block are stacked to form the temporal axis. Each of the $197 = 1 + 196$ output tokens (CLS + patch tokens) at a given hidden dimension constitutes one "neuron" — analogous to the feature-map $\times$ channel indexing used for convolutional models — yielding up to $197 \times 768 = 151{,}296$ candidate units per layer, of which 2000 are retained by max-activation sampling (see Figure 26).
- **Raptor**: The pipeline also supports the alternative framing in which transformer *layers* serve as the time axis (Jacobs et al., 2026); in that case single static frames (e.g. ImageNet images) are presented and activations are collected across all 12 encoder blocks, as done for the Raptor (Jacobs et al., 2026) model (see Figure 27).

B.2 DECODING MANIFOLD CONSTRUCTION

The decoding manifold embeds stimulus conditions as points in neural activity space.

Preprocessing. Prior to all analyses, each response tensor is shifted to be non-negative: $X \leftarrow X - \min_{\text{nan}}(X)$. Remaining NaN entries (variable-length padding for natural movie data) are replaced with 0 before PCA and metric computations.

Time-averaged manifold. Given the preprocessed population response tensor $X \in \mathbb{R}^{N\times S\times K\times T}$, we compute the time-averaged response matrix:

$$\hat{X} \in \mathbb{R}^{SK\times N}, \qquad \hat{X}_{(s,d),n} = \frac{1}{T}\sum_{t=1}^{T} X_{n,s,d,t}.$$

For datasets with NaN-padded variable-length stimuli, the mean is taken over valid (non-NaN) time bins only. PCA is applied to $\hat{X}$ to obtain a 3-component embedding $\mathbf{Z} = \text{PCA}_3(\hat{X}) \in \mathbb{R}^{SK\times 3}$ for visualization. k-NN accuracy, RSA, and CKA operate directly on the full-dimensional $\hat{X}$ without dimensionality reduction; Procrustes R^2 is computed in a 15-dimensional PCA space.

Time-resolved trajectories. For trajectory metrics, each time step is treated as a separate observation. The tensor is reshaped to $(T \cdot S \cdot D, N)$, PCA is fitted on all valid (non-NaN) rows, and the result is reshaped back to (SK, T, P) trajectories, where $P = 15$ PCA components. NaN-padded time steps remain NaN in the projected coordinates and are excluded from all trajectory metric computations. tRSA is computed on the full-dimensional space, other metrics on the PCA space.

B.2.1 STATIC DECODING MANIFOLD METRICS

Let $\hat{X}_1, \hat{X}_2 \in \mathbb{R}^{SK\times N}$ be the time-averaged response matrices of two neural populations.

k-NN Decoding Accuracy.

$$\text{Acc}(\hat{X}) = \frac{1}{SK}\sum_{i=1}^{SK} \mathbf{1}[\hat{y}_i = y_i], \qquad \hat{y}_i = \arg\max_c \sum_{j\in NN_k(i)} \mathbf{1}[y_j = c],$$

where y_i is the stimulus label of point i and $NN_k(i)$ is its leave-one-out k-nearest-neighbor set (k=5). Measures categorical separability.

Representational Similarity Analysis (RSA). Following Kriegeskorte et al. (2008), construct the representational dissimilarity matrix (RDM):

$$\mathbf{D}^1_{ij} = \left\|\bar{x}^1_i - \bar{x}^1_j\right\|_2, \qquad \mathbf{D}^2_{ij} = \left\|\bar{x}^2_i - \bar{x}^2_j\right\|_2,$$

where $\bar{x}_i = \frac{1}{T}\sum_t X_{:,i,t}$ is the time-averaged population response to condition i. The RSA score is the Spearman rank correlation between the two upper-triangular distance vectors:

$$\text{RSA} = \rho_s\big(\text{vec}(\mathbf{D}^1),\ \text{vec}(\mathbf{D}^2)\big) \in [-1, 1].$$

RSA measures whether the stimulus-similarity geometry of the two populations matches. RSA is sensitive only to ordinal structure, not absolute distances.

Linear CKA (Centered Kernel Alignment). Let $\mathbf{H} = \mathbf{I} - \frac{1}{m}\mathbf{1}\mathbf{1}^\top$ be the centering matrix ($m = SK$):

$$\text{CKA}(\hat{X}_1, \hat{X}_2) = \frac{\left\|\hat{X}_1^\top \mathbf{H}\hat{X}_2\right\|_F^2}{\left\|\hat{X}_1^\top \mathbf{H}\hat{X}_1\right\|_F \cdot \left\|\hat{X}_2^\top \mathbf{H}\hat{X}_2\right\|_F}.$$

CKA $\in [0, 1]$; invariant to orthogonal transformations and isotropic scaling (Kornblith et al., 2019; Williams et al., 2022).

Procrustes R^2. Let $\mathbf{Z}_1, \mathbf{Z}_2 \in \mathbb{R}^{SK \times P}$ denote their top-$P = 15$ PCA projections (for subpopulations of $k < 15$ neurons, $P = k$; the smaller manifold is zero-padded to 15 dimensions before alignment). We then solve the orthogonal Procrustes problem over the P-dimensional PCA embeddings (after mean-centering and unit-norm scaling $\rightarrow \tilde{\mathbf{Z}}$):

$$R^2_{\text{Proc}} = 1 - d^*_{\text{Proc}}, \qquad d^*_{\text{Proc}} = \min_{\mathbf{R} \in \mathcal{O}(P)} \left\| \tilde{\mathbf{Z}}_1 - \tilde{\mathbf{Z}}_2 \mathbf{R} \right\|_F^2.$$

$R^2_{\text{Proc}} \in [0, 1]$; values near 1 indicate the two point clouds are nearly congruent after optimal rigid alignment (Williams et al., 2022). Unlike CKA, Procrustes penalizes shape differences rather than alignment differences.

Sensitivity near ceiling. R^2_{Proc} compresses differences near 1, so we additionally recompute our main region-sampling (Section 4.2) and FPS (Section 4.3) contrasts with the Procrustes Frobenius distance $d_F = \|\tilde{Z}_1 - \tilde{Z}_2 R^*\|_F$ and the geodesic (principal) angle between the aligned subspaces, both of which are more sensitive than R^2 near ceiling.

Table 9: Procrustes alignment in three parameterizations (R^2 / d_F / θ) for the region-sampling (hi-/lo-PC1) and FPS (continuous/clustered) contrasts.

	hi-PC / lo-PC	continuous / clustered	Δ(cont$-$clust): R^2 / d_F / θ
Retina	0.952/0.220/12.6° / 0.410/0.768/45.2°	0.957/0.205/11.8° / 0.966/0.182/10.4°	-0.009 / $+0.024$ / $+1.4°$
V1	0.866/0.367/21.1° / 0.408/0.770/45.3°	0.865/0.367/21.1° / 0.880/0.346/19.9°	-0.015 / $+0.021$ / $+1.2°$
VISp (G)	0.909/0.302/17.4° / 0.584/0.645/37.6°	0.902/0.310/17.8° / 0.915/0.291/16.7°	-0.013 / $+0.019$ / $+1.1°$
VISp (NV)	0.989/0.105/6.0° / 0.701/0.546/31.7°	0.982/0.131/7.5° / 0.970/0.170/9.8°	$+0.012$ / -0.039 / $-2.2°$

The region-sampling (hi-PC vs. lo-PC) effect persists under both alternative measures (Δd_F in the range -0.34 to -0.55, $\Delta\theta$ in the range $-20°$ to $-33°$ across datasets), while the FPS (continuous vs. clustered) effect stays small and inconsistent in sign under d_F ($|\Delta d_F| \leq 0.04$) – consistent with the R^2-based conclusions in the main text.

B.2.2 Trajectory Metrics

tRSA operates directly in native neural space without any dimensionality reduction. SPC, tProc and DSA each project their population's time-resolved responses independently onto $P = 15$ PCA components (fitted separately per population). Let $\mathbf{z}^1_s(t) \in \mathbb{R}^P$ and $\mathbf{z}^2_s(t) \in \mathbb{R}^P$ denote the projected trajectory at time t for stimulus condition s (where applicable).

Time-Resolved RSA (tRSA). At each time bin t, compute instantaneous RDMs and their Spearman correlation:

$$\mathbf{D}^{1,2}_{ij}(t) = \left\| \mathbf{x}^{1,2}_i(t) - \mathbf{x}^{1,2}_j(t) \right\|_2, \qquad \text{RSA}(t) = \rho_s\big(\text{vec}(\mathbf{D}^1(t)),\ \text{vec}(\mathbf{D}^2(t))\big).$$

Summary scalar: $\overline{\text{tRSA}} = \frac{1}{T} \sum_t \text{RSA}(t)$.

Speed Profile Correlation (SPC).

$$v^{1,2}_s(t) = \left\| \mathbf{z}^{1,2}_s(t{+}1) - \mathbf{z}^{1,2}_s(t) \right\|_2, \qquad \rho^{\text{speed}}_s = \text{Pearson}\big(v^1_s(1{:}T{-}1),\ v^2_s(1{:}T{-}1)\big)\,.$$

Summary scalar: $\overline{\text{SPC}} = \frac{1}{SK} \sum_{s,k} \rho^{\text{speed}}_{s,k}$. Captures whether the *timing* of fast and slow phases of neural dynamics is preserved, even if absolute speeds differ.

Trajectory Procrustes R^2 (tProc). For each stimulus condition s, treat its T-step trajectory as a point cloud $\tilde{\mathbf{Z}}_s \in \mathbb{R}^{T \times P}$ (mean-centered and unit-norm-scaled) and apply orthogonal Procrustes alignment:

$$R^2_{\mathrm{Proc},s} = 1 - \min_{\mathbf{R}\in\mathcal{O}(P)} \left\|\tilde{\mathbf{Z}}^1_s - \tilde{\mathbf{Z}}^2_s\,\mathbf{R}\right\|^2_F.$$

Summary scalar: $\overline{R^2_{\mathrm{tProc}}} = \frac{1}{SK}\sum_{s,k} R^2_{\mathrm{Proc},s,k}$. This is the direct temporal analogue of Proc.: it measures whether the trajectory *shape* is preserved, not just the time-averaged endpoint.

Dynamical Similarity Analysis (DSA). Following Ostrow et al. (2023), DSA fits a linear dynamical system to each population's PCA-reduced trajectory via Dynamic Mode Decomposition (DMD) and computes the Procrustes distance between the learned transition operators $\mathcal{A}_1$ and $\mathcal{A}_2$:

$$\mathrm{DSA} = d_{\mathrm{DSA}}(\mathcal{A}_1, \mathcal{A}_2),$$

where d_{DSA} is the DSA distance. We report normalized (against the full population) z-scores of DSA values in comparison to 10 temporally shuffled baselines. Unlike other trajectory metrics, which compare trajectories geometrically, DSA asks whether the two populations implement the *same dynamical transformation* of stimuli over time (Appendix B.10).

B.3 Encoding manifold construction

The encoding manifold is constructed following the three-stage pipeline of Dyballa et al. (2024): (1) nonnegative tensor factorization to embed neurons into a stimulus-response space (Williams et al., 2018), (2) adaptive graph construction via IAN (Dyballa & Zucker, 2023), and (3) diffusion maps (Coifman & Lafon, 2006) for dimensionality reduction.

B.3.1 Preprocessing

The raw $(N \times S \times K \times T)$ tensor is preprocessed as follows. First, responses are smoothed along the time axis with a 1D Gaussian kernel ($\sigma_{\mathrm{NTF}} = 3$ bins). Next, for datasets with two spatial frequencies (11 stimuli), per-neuron optimal spatial frequency is selected based on population-level response magnitude, reducing the stimulus dimension to $S = 6$. The $K = 8$ directional responses are concatenated along the time axis to form a single response vector per (neuron, stimulus) pair of length $K \times T$. Finally, each neuron's response is normalized by its relative firing rate (divide by each neuron's maximum response, then rescale by relative activation). The resulting 3-tensor has shape $(N \times S \times (K \cdot T))$.

B.3.2 Nonnegative tensor factorization (NTF)

The preprocessed tensor $\mathbf{T} \in \mathbb{R}^{N\times S\times(DT)}$ is decomposed into R rank-1 nonnegative components (Williams et al., 2018):

$$\tilde{\mathbf{T}} = \sum_{r=1}^{R} \sigma_r^{\mathrm{NTF}}\, \mathbf{v}_r^{(1)} \circ \mathbf{v}_r^{(2)} \circ \mathbf{v}_r^{(3)},$$

where $\mathbf{v}_r^{(1)}$ are neural factors, $\mathbf{v}_r^{(2)}$ stimulus factors, and $\mathbf{v}_r^{(3)}$ temporal response factors. Factor vectors are normalized to unit length with magnitudes absorbed by σ_r^{NTF}. The decomposition is performed using the OPT method from Tensor Toolbox (Bader et al., 2023) with non-negativity constraints, run 50 times with different random initializations; the result with smallest reconstruction error is selected. The number of components R (typically 8–17) is chosen per dataset based on the explained variance heuristic of Dyballa et al. (2024). Circular permutations of the directional axis are applied during decomposition to detect response patterns irrespective of preferred orientation.

B.3.3 Neural encoding space

Following the reformulation in Dyballa et al. (2024), the scaled neural factor matrix $\mathcal{N}_{\sigma_{\mathrm{NTF}}} = \mathcal{N}\mathbf{\Sigma}_{\mathrm{NTF}}$ (where $\mathcal{N} = \mathbf{X}^{(1)}$ and $\mathbf{\Sigma}_{\mathrm{NTF}} = \mathrm{diag}(\sigma_1^{\mathrm{NTF}}, \ldots, \sigma_R^{\mathrm{NTF}})$) places each neuron as a point in an R-dimensional stimulus-response space. Distances in this space reflect similarity in tuning and temporal dynamics. PCA is applied to $\mathcal{N}_{\sigma_{\mathrm{NTF}}}$ to reduce dimensionality; the number of retained components is chosen so that cumulative explained variance exceeds 80%.

B.3.4 OUTLIER REMOVAL

Before graph construction, $n_{\text{far}} = 2$ most-isolated and $n_{\text{close}} = 5$ most-similar neurons (by minimum pairwise Euclidean distance) are removed as outliers.

B.3.5 ITERATED ADAPTIVE NEIGHBORHOODS (IAN)

On the PCA-reduced neural encoding space, a weighted data graph is constructed using the IAN algorithm (Dyballa & Zucker, 2023). IAN infers an adaptive local similarity kernel without requiring a fixed neighborhood size: it constructs the Gabriel graph, builds a multiscale Gaussian kernel, and iteratively prunes edges for consistency between discrete and continuous neighborhoods. Disconnected points identified by IAN are removed from subsequent analysis. The resulting sparse weighted adjacency matrix $\mathbf{K}$ encodes local similarities via locally tuned Gaussian kernels.

B.3.6 DIFFUSION MAPS

Diffusion maps (Coifman & Lafon, 2006) are applied to the IAN-weighted adjacency matrix $\mathbf{K}$. The matrix is normalized and symmetrized:

$$\mathbf{d}_i = \sqrt{\textstyle\sum_j \mathbf{K}_{ij} + \epsilon}, \qquad \mathbf{M}_s = \frac{\mathbf{K}}{\mathbf{d}\mathbf{d}^\top}.$$

The spectral decomposition of $\mathbf{M}_s$ yields eigenvalues $\mu_0 \geq \mu_1 \geq \cdots$ and eigenvectors $\boldsymbol{\psi}_l$. We compute $L = 20$ eigenvectors with diffusion time $t = 1$ and $\alpha = 1$, producing diffusion coordinates $\Psi_t(i) = (\mu_l^t \boldsymbol{\psi}_l(i))_{l=0}^{L-1}$.

B.3.7 MDS VISUALIZATION

For 3D visualization, classical MDS is applied to the pairwise squared Euclidean distances computed from the first diffusion coordinates: the Gram matrix $\mathbf{G} = -\frac{1}{2}\mathbf{H}\mathbf{D}^2\mathbf{H}$ (where $\mathbf{H}$ is the centering matrix) is eigendecomposed and the leading 10 components are retained.

B.4 ENCODING PIPELINE ROBUSTNESS

To validate that encoding manifold topology is stable across hyperparameters, we rerun the full encoding pipeline on V1 and Retina under all combinations of CP rank $F \in \{10, 15, 20\}$ and neuron fraction $\in \{50\%, 75\%, 100\%\}$, yielding 9 conditions per dataset (18 in total). The pipeline per condition is: response tensor $\rightarrow$ CP (5 restarts, max 200 iterations) $\rightarrow$ IAN $\rightarrow$ diffusion maps (20 components, $\alpha = 1$, $t = 1$) $\rightarrow$ MDS (10 components) $\rightarrow$ HDBSCAN (min_cluster_size $= 25$, min_samples $= 4$). Pairwise GW distances between all 18 embeddings (subsampled to 1200 points) are visualized as a distance heatmap and MDS projection (Figure 9).

Choice of embedding. We use diffusion maps (Section 3.2) rather than PCA to obtain the low-dimensional encoding embedding because it better preserves non-linear manifold structure (e.g. curved arms, local cluster boundaries) that a linear projection can distort. As a robustness check, we confirmed that the Figure 4 (continuous vs. clustered) GW contrast is qualitatively stable when PCA is substituted for diffusion maps in the embedding step, i.e. our conclusions are not an artifact of the specific non-linear embedding choice.

B.5 POPULATION-SIZE SWEEP (SECTION 4.1)

For each dataset, population fractions f drawn from 14 preset values $\{0.01, 0.02, 0.05, 0.10, 0.15, 0.20, 0.30, 0.40, 0.50, 0.60, 0.70, 0.80, 0.90, 1.00\}$, augmented by per-dataset absolute neuron counts $\{1, 2, 3, 5, 8, 13, 20\}$ (converted to fractions), are evaluated. At each fraction, $k = \lfloor f \cdot N \rfloor$ neurons are selected under six strategy types, each with high (top-k) and low (bottom-k) variants, yielding 11 distinct selection procedures:

1. **Random**: uniform random without replacement (10 seeds; mean and standard deviation reported).
2. **High/low curvature**: neurons ranked by RMS residual of a linear fit to the early Gaussian-smoothed response ($\sigma_{\text{smooth}} = 2$ bins, first $T_{\text{early}} = \min(40, T)$ time steps).

3. **High/low stability**: neurons ranked by mean absolute temporal derivative over the last T_{early} time steps (lower = more stable).
4. **High/low classifiability**: neurons ranked by variance of stimulus-averaged steady-state responses (last 10 time bins).
5. **High/low OSI**: neurons ranked by orientation selectivity index; skipped for single-direction datasets (natural movie, $D = 1$).
6. **High/low PC contribution**: neurons ranked by the L2-norm of their loading on the first global principal component of the full tensor reshaped to $(S \cdot K \cdot T) \times N$.

All eight metrics are computed for each subpopulation versus the full population.

B.6 Encoding manifold region sampling (Section 4.2)

For each dataset, two size-matched subpopulations are constructed by selecting the top-k (high-PC condition) or bottom-k (low-PC condition) neurons ranked by **PC contribution** — the L2-norm of each neuron's loading on the first global principal component of the decoding tensor reshaped to $(S \cdot K \cdot T) \times N$ (strategy 6 in Appendix B.5). Three fractions are evaluated: $f \in \{0.01, 0.05, 0.10\}$, giving $k = \lfloor fN \rfloor$.

All eight decoding metrics are computed vs. the full population. GW similarity between the subpopulation encoding manifold (first 3 MDS dimensions, ≤ 1200 points subsampled) and the full population is also computed.

The GW comparison panel (Figure 3J) pools GW values across all three fractions and four datasets (Retina, V1, VISp drifting gratings, VISp natural movie), yielding 12 matched pairs per condition (4 datasets $\times$ 3 fractions). Statistical significance is assessed with a Wilcoxon signed-rank test on matched pairs.

B.7 RSA Restricted to Principal-Component Bands

As an additional robustness check on the FPS (continuous vs. clustered, Section 4.3) and region-sampling (hi-PC1 vs. lo-PC1, Section 4.2) contrasts, we recompute RSA after restricting the decoding-space PCA reconstruction to specific principal-component bands, rather than using the full aggregate. This decomposes the aggregate RSA difference into a per-band signal and tests whether it is concentrated in a particular part of the spectrum. The mid band (PC4–15) is the most discriminative band for this contrast in three of four datasets, but its *sign* is not reliable across datasets (positive for Retina, VISp (G), and VISp (NV), negative for V1), so a PC-band decomposition of RSA does not, on its own, provide a dataset-general diagnostic; it remains a decoding-space-only quantity and does not substitute for the encoding-manifold/GW comparison in the main text.

Table 10: RSA(continuous) − RSA(clustered), restricted to PC bands of the decoding space.

	PC1	PC1–3	PC4–15	PC16+
Retina	−0.001	−0.064	**+0.129**	−0.033
V1	−0.055	−0.080	**−0.153**	−0.155
VISp (G)	−0.222	+0.056	**+0.352**	+0.010
VISp (NV)	+0.015	+0.010	**+0.080**	–

B.8 Farthest Point Sampling (Section 4.3)

FPS constructs subpopulations with controlled topological coverage of the encoding manifold:

1. **Seed selection.** The first seed is chosen uniformly at random from the encoding manifold embedding (first 3 MDS dimensions). Subsequent seeds are selected greedily by the farthest-point-sampling criterion: at each step, the neuron with the largest minimum Euclidean distance to all previously selected seeds is added. This produces n seeds that maximally cover the manifold.

2. **Nearest-neighbor expansion.** For each seed, its m nearest neighbors in the full MDS embedding (all 10 dimensions) are added to the subpopulation, excluding neurons already selected. The resulting subpopulation contains at most $n \cdot (m + 1)$ neurons.
3. **Conditions.** Two canonical conditions are compared:
   - $(n{=}50, m{=}1)$: 50 widely spread seeds, each with one neighbor $\rightarrow$ 100 neurons spanning the manifold smoothly.
   - $(n{=}10, m{=}9)$: 10 seeds, each with 9 neighbors $\rightarrow$ 100 neurons grouped into 10 tight clusters.

   Both conditions produce subpopulations of $\approx$ 100 neurons but with opposite encoding manifold topologies. Five random seeds are used for the initial seed selection: $\{42, 43, 44, 45, 46\}$. Mean and standard deviation across seeds are reported as central estimate and error bars in Figure 4.

All eight metrics are computed for each FPS subpopulation versus the full population.

B.9 MNIST CAUSAL DISSOCIATION EXPERIMENT (SECTION 4.4)

B.9.1 ARCHITECTURE

We train a three-layer convolutional neural network on MNIST digit classification:

- `Conv1`: $1 \rightarrow 16$ channels, 3×3 kernel $\rightarrow$ ReLU $\rightarrow$ MaxPool$(2) \rightarrow (B, 16, 13, 13)$
- `Conv2`: $16 \rightarrow 32$ channels, 3×3 kernel $\rightarrow$ ReLU $\rightarrow$ MaxPool$(2) \rightarrow (B, 32, 5, 5)$
- Flatten $\rightarrow (B, 800)$ (*"neural population"*)
- FC: $800 \rightarrow 10$ (logits)

Each of the $N = 800$ units in the flattened layer corresponds to a (channel, spatial position) pair in the 32 feature maps of `Conv2`. This layer is treated as the neural population for all encoding and decoding analyses.

B.9.2 TRAINING

All models are trained with Adam (learning rate 10^{-3}, batch size 256) for 5 epochs on the standard MNIST training set ($\sim$ 60 000 images). The standard MNIST test set (10 000 images) is used for evaluation. We use 20 seeds for initialization to obtain statistically robust results.

B.9.3 BASELINE CONDITION

The baseline model is trained with cross-entropy loss only: $\mathcal{L} = \mathcal{L}_{\text{CE}}$.

B.9.4 CLUSTERING LOSS ($\mathcal{L}_{\text{CLUSTER}}$)

For each value of $\lambda \in \{0.1, 0.5, 1, 5, 10, 50\}$, the model is trained with:

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda \cdot \mathcal{L}_{\text{cluster}}.$$

The clustering loss $\mathcal{L}_{\text{cluster}}$ is a supervised contrastive loss operating on the activation profiles of the 800 neurons across the batch. For each neuron, its response vector across the B samples in the current mini-batch serves as a high-dimensional feature. Neurons are labeled by their *preferred stimulus* (the digit class eliciting the highest mean activation). The SupCon (Khosla et al., 2021) loss then pulls neurons with the same preferred stimulus together and pushes neurons with different preferences apart:

$$\mathcal{L}_{\text{cluster}} = \frac{1}{|\mathcal{A}|} \sum_{i \in \mathcal{A}} \frac{1}{|P(i)|} \sum_{p \in P(i)} -\log \frac{\exp(\mathbf{z}_i \cdot \mathbf{z}_p / \tau)}{\sum_{j \neq i} \exp(\mathbf{z}_i \cdot \mathbf{z}_j / \tau)},$$

where $\mathbf{z}_i = \mathbf{h}_i / \|\mathbf{h}_i\|$ is the ℓ_2-normalized activation profile of neuron i (a B-dimensional vector), $P(i) = \{j \neq i : \text{pref}(j) = \text{pref}(i)\}$ is the set of neurons sharing neuron i's preferred digit,

$\mathcal{A} = \{i : |P(i)| > 0\}$ is the set of anchor neurons with at least one positive, and $\tau = 0.07$ is the temperature.

This loss directly clusters the encoding manifold—neurons responding similarly to the same digit class are pulled toward shared prototypes—while the CE term ensures that task accuracy is maintained.

B.9.5 EVALUATION TENSOR

For each trained model, a $(800 \times 10 \times 50 \times 1)$ response tensor is constructed: the first 50 test images of each digit class (test-set order), with the 800-dimensional hidden activation recorded for each. The singleton fourth dimension ($T = 1$) is a placeholder for time, ensuring compatibility with the subpopulation analysis pipeline. Stimulus labels are $\mathbf{y} = (0, 0, \ldots, 0, 1, 1, \ldots, 9, 9)$ with 50 entries per class.

B.9.6 ENCODING MANIFOLD CONSTRUCTION

For each condition, the encoding manifold is built via the full CP $\rightarrow$ IAN $\rightarrow$ diffusion maps $\rightarrow$ MDS pipeline (Appendix B.3), with the following MNIST-specific settings:

- CP rank $R = 10$, with a single direction ($K = 1$, no directional tuning). 5 random restarts of the CP optimizer.
- IAN is run on the R-dimensional neural encoding matrix with a small jitter ($\gamma = 10^{-4}$) added to break distance ties.
- Diffusion maps: $L = 20$ eigenvectors, $\alpha = 1$, $t = 1$.
- MDS: 5 components (first 2 used for 2D visualization).

B.9.7 ENCODING SIMILARITY (GROMOV–WASSERSTEIN)

Encoding manifold similarity is quantified via the Gromov–Wasserstein (GW) distance, solved exactly without entropic regularization (Python Optimal Transport library, `ot.gromov.gromov_wasserstein2`, `square_loss`, max 10 000 iterations, tolerance 10^{-4}).

B.9.8 ATTRIBUTION ANALYSIS

We use gradient$\times$activation attribution to quantify which neurons drive class-specific outputs.

FC layer (800 neurons). For each digit class c and model seed, we pass $n = 200$ correctly classified test examples through the network, record the 800-dimensional hidden activation $\mathbf{h}$, and compute $\mathrm{attr}_{i,c} = |\partial \ell_c / \partial h_i| \cdot h_i$, where ℓ_c is the class-c logit. Averaging over examples yields a (10×800) attribution matrix.

Conv2 layer (32 channels). The same procedure is applied to Conv2 pre-activations, spatially average-pooled over the 5×5 feature map, using $n = 300$ examples per class, yielding a (10×32) matrix.

Entropy. For class c, normalize absolute attributions: $\mathbf{p}_c = |\mathbf{a}_c| / \|\mathbf{a}_c\|_1$. Attribution entropy: $H_c = -\sum_i p_{c,i} \log p_{c,i}$; report $\bar{H} = \frac{1}{10} \sum_c H_c$. High entropy indicates distributed attribution across neurons; low entropy indicates concentration on a few.

Overlap. Mean cosine similarity between normalized attribution vectors across all 45 class pairs: $\bar{\cos} = \mathrm{mean}_{c \neq c'}(\hat{\mathbf{a}}_c \cdot \hat{\mathbf{a}}_{c'})$. High overlap indicates similar attribution patterns across classes (shared circuits); low overlap indicates class-specific circuits.

Assortativity. An undirected graph is built on neurons (FC: 800 nodes; Conv2: 32 channel nodes). Each node is labeled by its preferred digit: for FC neurons, the argmax of the mean tuning response; for Conv2 channels, the mode of neuron preferences within the channel. An edge (u, v) exists iff the cosine similarity between L2-normalized weight vectors exceeds a threshold ($\theta_{\mathrm{FC}} = 0.5$;

$\theta_{\text{Conv2}} = 0.3$). Assortativity is computed via NetworkX. Positive values indicate that same-class neurons preferentially connect within class-specific pathways.

B.9.9 STATISTICAL SIGNIFICANCE

We apply Wilcoxon signed-rank tests (paired by seed) to accuracy and blur robustness, and Wilcoxon rank-sum tests to all other metrics, with Holm–Bonferroni correction, to assess the statistical significance of differences between the MNIST-trained models.

B.10 DYNAMICAL SIMILARITY ANALYSIS (DSA)

DSA (Ostrow et al., 2023) is computed as follows. The response tensor is reduced to $P = \min(15, N)$ PCA components per population (randomized SVD, fitted independently for reference and subpopulation; NaN-padded frames replaced with zero). The DSA package fits a linear dynamical system via Dynamic Mode Decomposition (DMD) to each population's trajectories (shape (SK, T, P)) with $n_{\text{delays}} = \min(\lfloor T/2 \rfloor, 20)$ delay-embedding steps, adaptive rank $\text{rank} = \min(P \cdot n_{\text{delays}}, 2P)$, and no Tikhonov regularization (package default).

Normalization. A null distribution is obtained from $n_{\text{shuf}} = 10$ temporal permutations of the subpopulation trajectories, giving null mean $\bar{d}_{\text{shuf}}$ and std σ_{shuf}. The z-score $z = (\bar{d}_{\text{shuf}} - d)/\sigma_{\text{shuf}}$ is divided by the full-population self-z-score z_{ref} (computed once per dataset) so that the full population scores exactly 1: $\text{DSA} = \text{clip}(z/z_{\text{ref}}, 0, 1)$. Implementation via the `dsa-analysis` package.

B.11 IMPLEMENTATION DETAILS

All experiments are implemented in Python using NumPy (Harris et al., 2020), SciPy (Virtanen et al., 2020), scikit-learn (Pedregosa et al., 2011), and PyTorch (Paszke et al., 2019). The IAN algorithm uses the implementation of Dyballa & Zucker (2023) (v1.1.2). Diffusion maps use the `diffusionMapSparseK` function from the IAN package. Gromov–Wasserstein distances are computed with the Python Optimal Transport library (Flamary et al., 2021). DSA uses the `dsa-analysis` package. Visualizations use Matplotlib (Hunter, 2007).

All software (Table 11) is used in accordance with its respective license.

Table 11: Software packages used in this work.

Package	**Version**	**License**
MATLAB Tensor Toolbox (Bader et al., 2023)	3.6	BSD-2
IAN (Dyballa & Zucker, 2023)	1.1.2	BSD-3
NumPy (Harris et al., 2020)	1.25.0	BSD-3
SciPy (Virtanen et al., 2020)	1.15.3	BSD-3
scikit-learn (Pedregosa et al., 2011)	1.7.1	BSD-3
PyTorch (Paszke et al., 2019)	2.6.0	MIT
Matplotlib (Hunter, 2007)	3.10.1	PSF-based (BSD-compatible)
Plotly (Inc., 2015)	6.0.0	MIT
TUEplots (Krämer et al., 2024)	0.2.0	MIT

B.12 REPRODUCIBILITY

All random seeds are fixed and reported: FPS sweep seeds = $\{42, 43, 44, 45, 46\}$, MNIST random initialization seeds = 0–19. Code is available at `https://anonymous.4open.science/r/Decoding_subpopulations-D717/README.md`, and we provide an interactive Neural Manifold Explorer tool.