Decoding Alignment without Encoding Alignment: A Missing Component for Interpretability

Johannes Bertram [1 2 3] **Luciano Dyballa** [4] **T. Anderson Keller** [5] **Savik Kinger** [6] **Steven W. Zucker** [6 7 8]

Abstract

RSA and CKA are standard metrics for comparing neural representations across brain regions, organisms, and deep learning models. We demonstrate a fundamental weakness: these decoding-based metrics are insensitive to *encoding manifold topology* — the internal functional organization of a neural population. In a controlled MNIST experiment, RSA, CKA, and Procrustes R^2 remain statistically unchanged when encoding topology is causally manipulated via an auxiliary clustering loss, while the two model populations differ significantly in attribution patterns, weight-graph assortativity, and out-of-distribution robustness. Across biological systems and machine learning models, similar decoding behavior can arise from small, non-representative subpopulations, and alignment metrics are insensitive to encoding manifold topology even when it is fundamentally altered. These findings bear directly on mechanistic interpretability: standard alignment metrics cannot distinguish whether two networks share the same computational circuits or merely produce indistinguishable aggregate outputs. We propose encoding manifolds and Gromov–Wasserstein distance as complementary diagnostics for any decoding-based similarity claim, and provide a Neural Manifold Explorer tool for further experimentation.

Presented at the Mechanistic Interpretability Workshop at ICML 2026. [1]University of Tübingen, Germany [2]ELLIS Institute Tübingen, Germany [3]Tübingen AI Center, Germany [4]School of Science & Technology, IE University, Madrid, Spain [5]The Kempner Institute for Natural and Artificial Intelligence, Harvard University, Cambridge MA, USA [6]Department of Computer Science, Yale, New Haven CT, USA [7]Department of Biomedical Engineering, Yale, New Haven CT, USA [8]Wu Tsai Institute, Yale, New Haven CT, USA. Correspondence to: Johannes Bertram <jb@w3a.de>.

Proceedings of the 43rd International Conference on Machine Learning, Seoul, South Korea. PMLR 306, 2026. Copyright 2026 by the author(s).

1. Introduction

A central goal in interpretability and 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; Cadieu et al., 2014). The dominant approach records population responses to shared stimuli and compares them via scalar alignment scores such as RSA (Kriegeskorte et al., 2008) and CKA (Kornblith et al., 2019). While useful for decoding, scores over states offer only a partial view of computational organization: Two systems that are not "digital twins" can achieve indistinguishable alignment scores — yet the same population state can be produced by qualitatively different internal functional organizations. The same challenge is central to mechanistic interpretability (Ameisen et al., 2025; Lindsey et al., 2024): two models trained differently, pruned, or fine-tuned may match on RSA or CKA while implementing behavior through entirely different internal circuits.

The reason is that the map from stimuli to behavior via population states is underspecified: many different internal organizations can produce the same population-level distance geometry (Ganguli & Sompolinsky, 2012; 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 state and examine *how* neurons (or units) are internally organized. We do so using the *encoding manifold* (Dyballa et al., 2024; Posani et al., 2025): rather than representing stimuli as points in neural-response space, the encoding manifold represents *neurons* as points in stimulus-response space. 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, independently of any downstream readout (Figure 1).

Decoding metrics treat the neural population as a monolithic input/output 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:

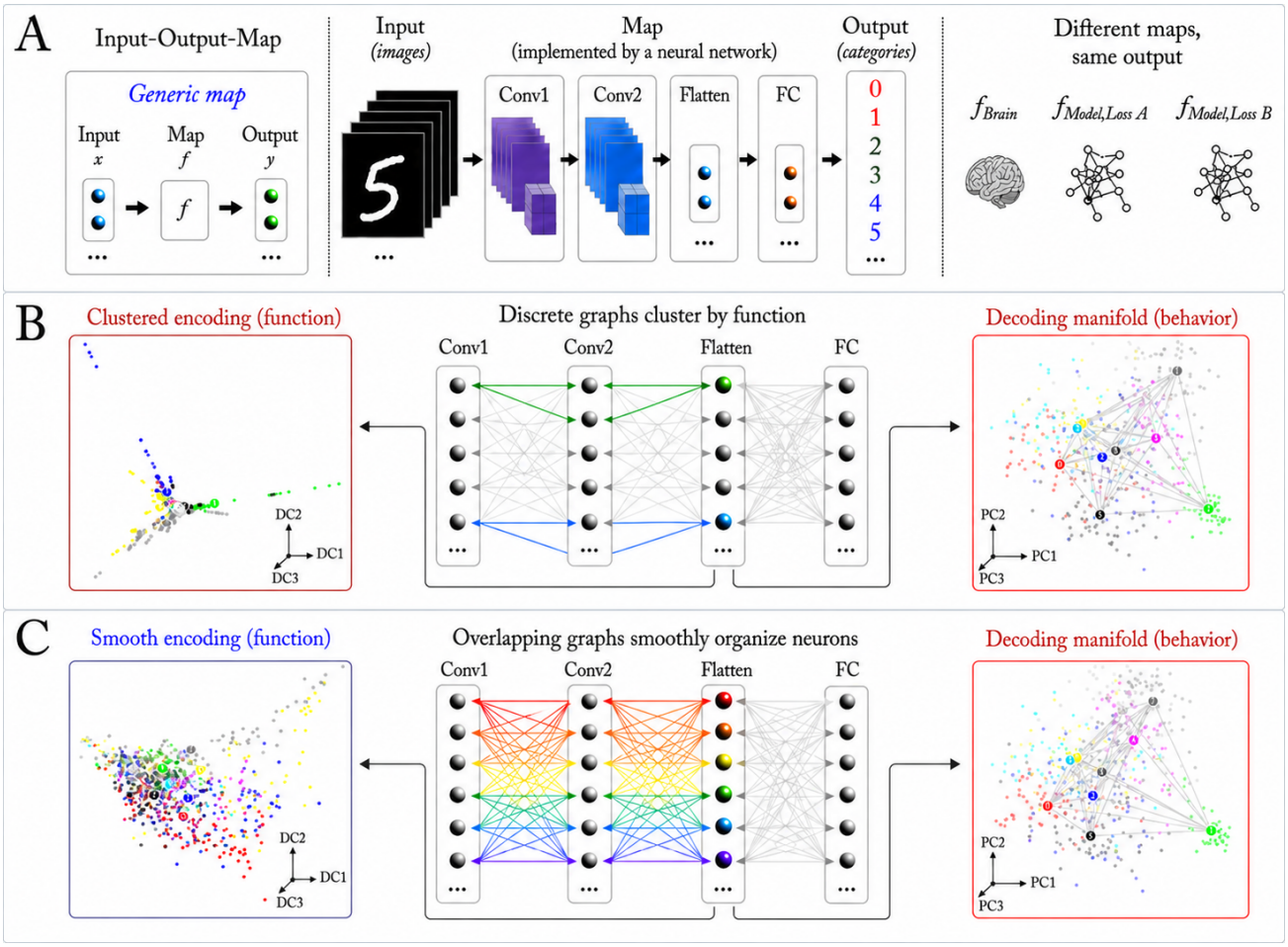

Figure 1. **Encoding and decoding manifolds provide complementary views of neural systems**. (A) A neural network as an input-output map. The same behavior – categorization – can be achieved with different maps. (B) Stimuli can be represented in neural coordinates (decoding manifold, right, colors represent different stimuli) or neurons can be represented in stimulus coordinates (encoding manifold, left, colors represent neuron's preferred stimulus). A clustered decoding manifold implies discrimination between stimulus classes; a clustered encoding manifold (non-overlapping preferred stimulus 'arms') implies separation between neuron's 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. This artificial example is developed in Sec. 3.1.

1. A small, selective subset of neurons can replicate the full population's decoding performance and may achieve this through *different* neural subpopulations.
2. Representation 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 by design, providing *causal* rather than merely correlational evidence that decoding metrics are insensitive to internal functional organization.

We demonstrate these findings across mouse retina and V1, five Allen cortical areas, and several machine learning models (Appendix D.1). 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 encoding manifolds and Gromov-Wasserstein distance as complementary diagnostics (Dyballa et al., 2024; Bertram et al., 2026); see Appendix A for extended discussion of prior work.

2. Methods

2.1. 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 and therefore do not confound with subpopulation

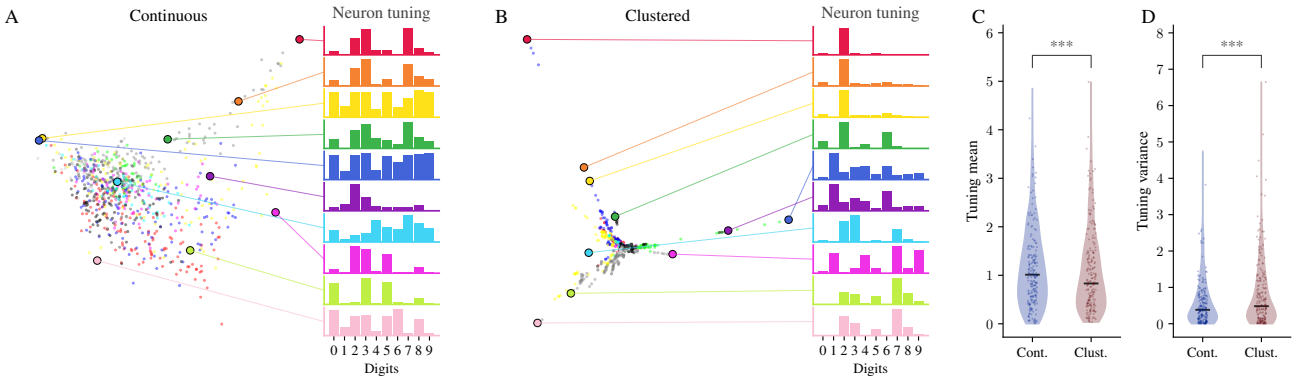

Figure 2. MNIST experiment: continuous (A) vs. clustered (B) encoding manifold. Tuning histograms show graded, multi-class responses in A and sharp, single-class peaks in B. Neurons in the clustered model have significantly lower tuning mean (C) and higher variance (D), indicating sharper class-selective responses. (Wilcoxon rank-sum; * $p<0.05$, ** $p<0.01$, *** $p<0.001$)

Table 1. MNIST results (n=20 seeds, mean ± SEM). **Top:** decoding and encoding metrics for λ=10 vs. baseline; cross-condition metrics relative to baseline. **Bottom:** functional organization metrics for λ=10 vs. baseline (Wilcoxon rank-sum; * $p<0.05$, *** $p<0.001$, with Bonferroni-Holm correction.)

Metric	Baseline (λ=0)	Clustered (λ=10)	Sig.
RSA	0.958 ± 0.001	0.961 ± 0.002	ns
CKA	0.976 ± 0.001	0.976 ± 0.001	ns
Proc. R^2	0.950 ± 0.002	0.941 ± 0.008	ns
GW Sim (ours)	**0.958** ± 0.001	**0.502** ± 0.022	***
Functional Differences			
Blur robustness, mean acc. ($\sigma_{\text{blur}} \geq 4$)	0.318 ± 0.010	0.282 ± 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 ± 0.003	***

size, and constructed such that high values mean higher alignment. Definitions and implementation details are given in Appendix D.2.1.

2.2. Encoding Manifold

The *encoding manifold* (Dyballa et al., 2024; Bertram et al., 2026) describes how neurons are organized in *neural response space* rather than stimulus space. The encoding manifold is the point cloud $\mathcal{E}$; 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. We construct it via nonnegative tensor factorization → IAN adaptive graph (Dyballa & Zucker, 2023) → diffusion maps (Coifman et al., 2005; Coifman & Lafon, 2006), following Dyballa et al. (2024); full details are given in Appendix D.3. The encoding manifold is biologically grounded: applied to retina it recovers the known retinal ganglion cell types as distinct clusters, whereas mouse V1 — which has no known discrete functional types — forms a smooth continuum (Dyballa et al., 2024; Nellen et al., 2025). This validates it as a descriptor of functional organization rather than an arbitrary embedding.

We measure encoding manifold similarity via Gromov–Wasserstein (GW) distance (Peyré & Cuturi, 2020; Mémoli, 2011), applied here as an intrinsic measure of encoding manifold coverage rather than cross-system alignment. GW measures how well the internal pairwise-distance structure of one point cloud can be matched to another: two manifolds with the same shape score zero, while structurally incompatible ones score high. Normalized GW similarity is:

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

$$\mathrm{GW}_{\mathrm{Sim}}(\mathcal{E}_{\mathrm{sub}}) = 1 - \frac{\mathrm{GW}(\mathcal{E}_{\mathrm{sub}},\, \mathcal{E}_{\mathrm{full}})}{\mathrm{GW}_{\max}},$$

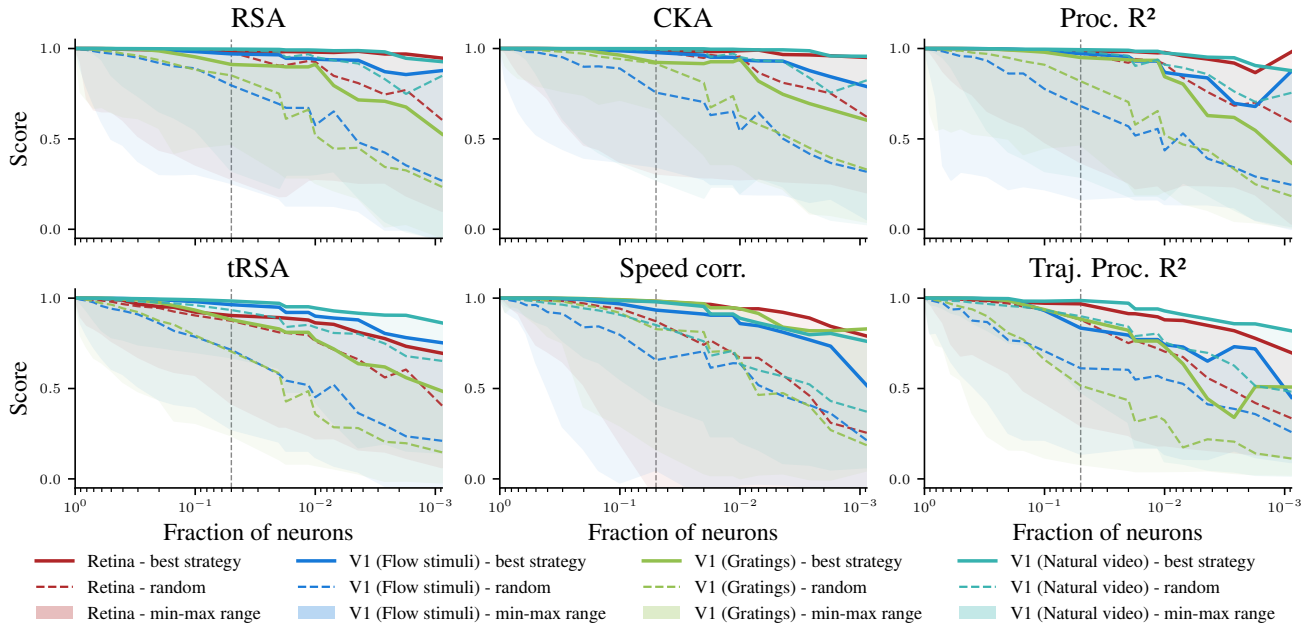

Figure 3. Population-size sweep across Retina, V1 (flows, gratings, natural) for six decoding metrics. Solid lines: best-performing selection strategy; dashed lines: random selection; shaded bands: min–max range across all strategies. The vertical dashed line marks 5% of neurons (used in Figure 9). Static metrics plateau near ceiling at 5% or below under the best strategy.

where $\mathcal{E}_{\text{sub}}$ and $\mathcal{E}_{\text{full}}$ are encoding manifolds of the subpopulation and full population, and $\text{GW}_{\max} = \text{GW}(\mathcal{E}_{\text{full}}, \mathcal{E}_{\text{coll}})$ normalizes by a collapsed-centroid baseline ($\epsilon = 10^{-8}$ jitter to prevent degeneracy).

3. Results

3.1. Causal Dissociation via Controlled Training

We train a three-layer CNN on MNIST digit classification (10 classes, $\approx$ 60 000 images; details in Appendix D.8) and add an auxiliary loss $\mathcal{L}_{\text{cluster}}(\lambda)$ that progressively pulls hidden-layer neurons toward digit-specialist response templates, systematically clustering the encoding manifold (Khosla et al., 2021).

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

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. Task accuracy is held approximately constant ($\approx 98.5\%$) across all values of λ.

Table 1 reports decoding metrics and GW encoding similarity as λ increases from 0 to 50. GW similarity drops monotonically from 1.00 to 0.28 — the previously continuous population response space becomes sharply clustered around digit-specialist prototypes (Figure 2) — while RSA, CKA, and Procrustes R^2 remain above 0.93 up to λ=10. We focus on λ=10 vs. baseline (20 seeds each) for the main analysis.

Functional Differences. Despite matched decoding behavior, the two models differ functionally. Neurons in the clustered model show significantly lower mean tuning and higher variance (Figure 2 C, D), reflecting sharper, class-selective responses (B) vs. graded, multi-class tuning (A). Gradient-activation attribution (Table 1; Appendix D.8.8) reveals that the constrained model uses more class-specific circuits: attribution entropy and overlap are significantly lower (p<0.001) and weight-graph assortativity significantly higher (p<0.001), indicating same-class neurons preferentially connect within class-specific pathways. Out-of-distribution blur robustness also differs significantly (p<0.05). Together, two models indistinguishable by RSA, CKA, or task accuracy implement their shared behavior through fundamentally different internal architectures.

3.2. Decoding Metrics Saturate with Small Subpopulations

We next show that decoding scores can be reproduced by a small fraction of neurons. We perform a population-size sweep under six neuron-selection strategies (Appendix D.5). Across biological systems (Figure 3), all metrics saturate

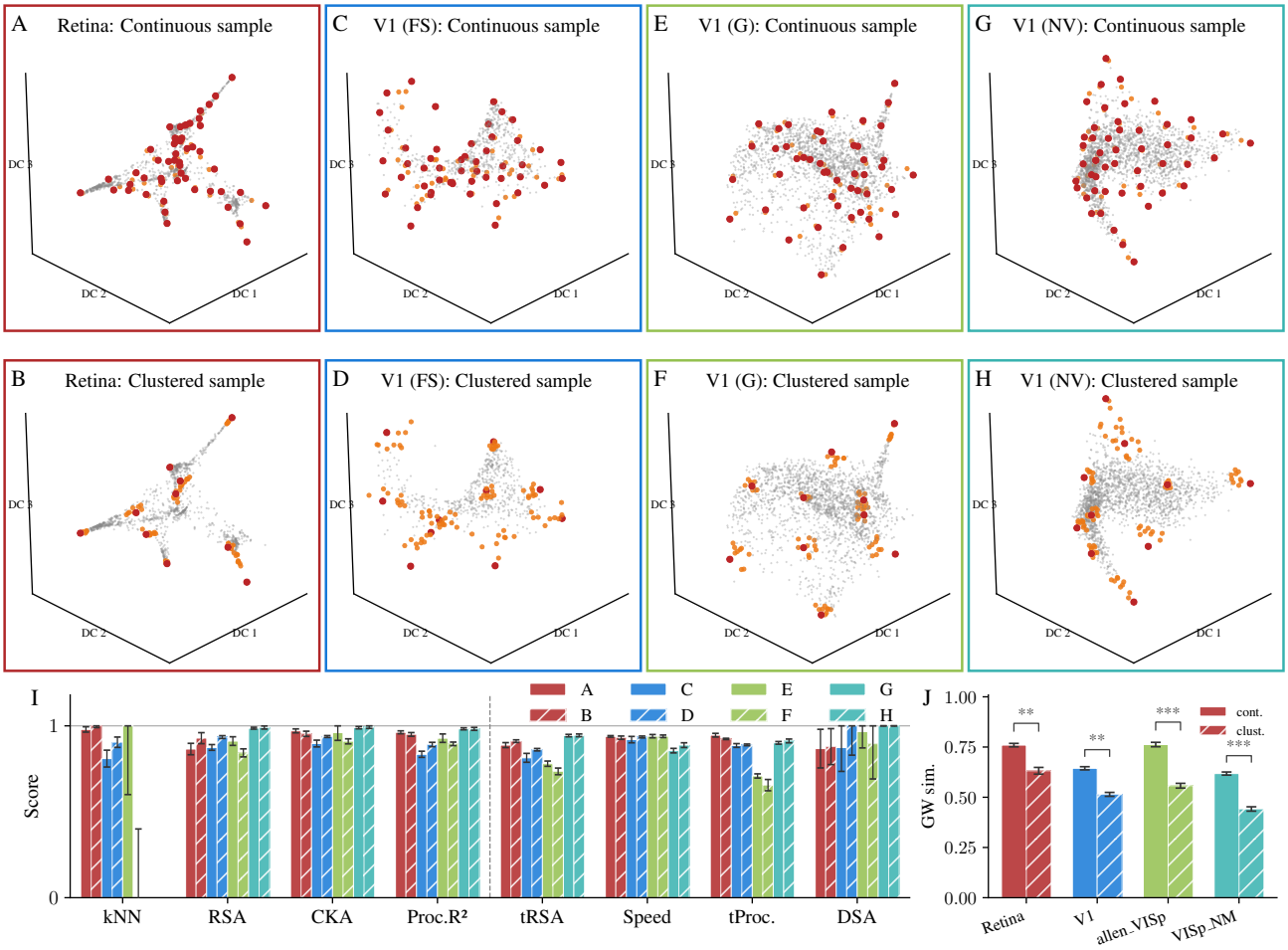

Figure 4. FPS-based subpopulations with continuous (top row, dark red) vs. clustered (bottom row, orange) encoding manifold topology for Retina, V1, VISp (gratings), and VISp (natural). Continuous subpopulations are sparsely distributed across the full manifold; clustered subpopulations are concentrated in local neighborhoods. Bar chart shows all eight metrics for both conditions; error bars are 1 s.d. over FPS seeds. Scores are near-ceiling for both topologies across nearly all systems and metrics, demonstrating that decoding metrics are insensitive to this fundamental difference in encoding structure. (Wilcoxon rank-sum; ** $p<0.01$, *** $p<0.001$)

near ceiling at or above 5% of neurons under information-maximizing strategies; RSA and CKA remain near ceiling even under random selection. This motivates asking what the remaining neurons implement — the following experiment tests this directly.

3.3. Encoding Manifold Topology Can Be Disrupted While Preserving Decoding Metrics

We use *Farthest Point Sampling* (FPS) to construct subpopulations (Appendix D.7): by varying n seed neurons (breadth of manifold coverage) and m nearest neighbors per seed, we interpolate between a **continuous** sample that spans the full encoding manifold (m=1) and a locally **clustered** sample (m>1). Figure 4 shows results for Retina, V1, and Allen V1 (VISp).

The correlation-based decoding metrics remain near-ceiling for both functional topologies across systems (Figure 4). Together, Retina, V1, and ten Allen cortical datasets (five areas × two stimulus classes) establish that the insensitivity of RSA, CKA, DSA and other decoding metrics to encoding manifold topology is robust and general. The one exception is k-NN accuracy, a hard classifier rather than a correlation: it can drop sharply for clustered samples (e.g. VISp gratings, 0.80 → 0.20 in Table 4). Machine learning models including FNN, Flyvision, and R(2+1)D show the same pattern (Appendix C.2.2). Full per-dataset scores and manifold visualizations for all 17 datasets are given in Table 4 and Figures 12–26.

Where all these decoding metrics fail to differentiate, our GW similarity provides a significant distinction between continuous and clustered samples in all settings (Figure 4 J). The continuous sample always spans the full encoding manifold better and is thus measurably more similar in function to the full population. GW similarity thus provides a complementary, encoding-side measure that captures precisely

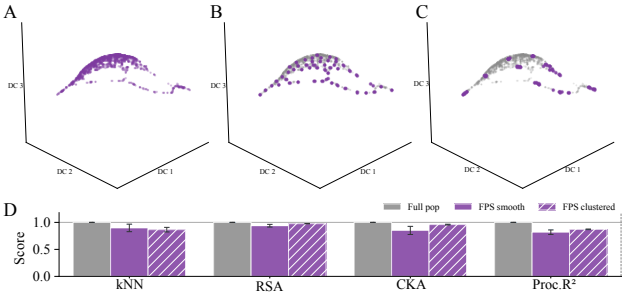

Figure 5. FPS-based subpopulations with continuous (B) and clustered (C) encoding manifold topology for a ViT. Stationary decoding metrics are high across conditions and indistinguishable between the two encoding manifold topologies.

what decoding metrics are blind to. The region-sampling experiment (Appendix C.1) further shows that encoding manifold position predicts decoding fidelity, ruling out redundancy as an explanation.

We provide a Neural Manifold Explorer tool for further experimentation: an interactive 3-D viewer for inspecting the encoding and decoding manifolds and subpopulations of every system analyzed here.

3.4. Vision Transformers

Our pipeline assumes a neuron $\times$ stimulus $\times$ direction $\times$ time response tensor, but a vision transformer (ViT) (Dosovitskiy et al., 2021) has neither a canonical neuron axis nor a time axis. We obtain our data tensor from relating hidden dimensions and tokens in the ViT to channels and within-channel neurons in CNNs: We map a ViT onto our framework by treating each (token, hidden-dimension) entry as a "neuron" (up to 197×768 candidate units, of which 2000 are retained by max-activation sampling), and by supplying the temporal axis either from feeding frame sequences through a fixed encoder block or from treating successive encoder layers as time (the latter "layers-as-time" framing is detailed for the Raptor model; Appendix D.1, Figures 27, 28). With this mapping, the same pattern reproduces (Figure 5, Table 4): correlation-based decoding metrics remain high and statistically indistinguishable between FPS-continuous and FPS-clustered encoding topologies. We make no claim here about the specific structure of ViT encoding manifolds; rather, this demonstrates that the encoding-manifold diagnostic is architecture-agnostic and can be applied to transformer models.

4. Discussion

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 3.1). We therefore recommend complementing decoding-based comparisons with encoding manifolds and GW-based measures of functional organization.

Implications for mechanistic interpretability. Circuit-tracing methods ask which neurons and pathways implement specific behaviors (Ameisen et al., 2025; Lindsey et al., 2024; Dunefsky et al., 2024). Our results show that standard alignment metrics are blind to precisely what circuit analysis seeks to reveal: the internal functional organization of a population. Our MNIST experiment makes this concrete: models statistically indistinguishable by RSA and CKA differ significantly in tuning, attribution structure, and weight-graph assortativity. We do not claim the encoding manifold identifies causal circuits; rather, it provides a scalable, geometric complement — by organizing neurons by response profiles rather than by stimuli, it surfaces differences in functional organization that decoding metrics miss. We propose encoding manifolds and GW-based similarity as scalable complements to circuit tracing for assessing functional equivalence, not merely behavioral equivalence.

Conclusion. 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 gives a principled way to compare it across systems. 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.

5. Acknowledgments and Disclosure of Funding

This article has received funding from the European Commission's Marie Skłodowska-Curie Action under grant agreement no. 101207931 (LD), and by the MICIU /AEI /10.13039/501100011033 / FEDER, UE Grant no. PID2024-155187OB-I00 (LD). SWZ was supported by NIH grant 1R01EY031059 and NSF Grant 1822598.

References

Ahlert, J., Klein, T., Wichmann, F., and Geirhos, R. How Aligned are Different Alignment Metrics?, 2024. URL http://arxiv.org/abs/2407.07530.

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

Ameisen, E., Syed, M., Lindsey, J., Tamkin, A., Templeton, A., Marcus, J., Conerly, T., Batson, J., and Olah, C. Circuit tracing: Revealing computational graphs in language models. *Transformer Circuits Thread*, 2025. URL https://transformer-circuits.pub/2025/attribution-graphs/methods.html.

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

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

Bertram, J., Dyballa, L., Keller, A., Kinger, S., and Zucker, S. W. How 'Neural' is a Neural Foundation Model?, 2026. URL http://arxiv.org/abs/2601.21508.

Bo, Y., Soni, A., Srivastava, S., and Khosla, M. Evaluating Representational Similarity Measures from the Lens of Functional Correspondence, 2025. URL http://arxiv.org/abs/2411.14633.

Cadieu, C. F., Hong, H., Yamins, D. L. K., Pinto, N., Ardila, D., Solomon, E. A., Majaj, N. J., and DiCarlo, J. J. Deep Neural Networks Rival the Representation of Primate IT Cortex for Core Visual Object Recognition, 2014. URL https://arxiv.org/abs/1406.3284v1.

Coifman, R. R. and Lafon, S. 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.

Coifman, R. R., Lafon, S., Lee, A. B., Maggioni, M., Nadler, B., Warner, F., and Zucker, S. W. 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.

Davari, M., Horoi, S., Natik, A., Lajoie, G., Wolf, G., and Belilovsky, E. Reliability of CKA as a Similarity Measure in Deep Learning, 2022. URL http://arxiv.org/abs/2210.16156.

de Vries, S. E. J., Lecoq, J. A., Buice, M. A., Groblewski, P. A., Ocker, G. K., Oliver, M., Feng, D., Cain, N., Ledochowitsch, P., Millman, D., Roll, K., Garrett, M., Keenan, T., Kuan, L., Mihalas, S., Olsen, S., Thompson, C., Wakeman, W., Waters, J., Williams, D., Barber, C., Berbesque, N., Blanchard, B., Bowles, N., Caldejon, S. D., Casal, L., Cho, A., Cross, S., Dang, C., Dolbeare, T., Edwards, M., Galbraith, J., Gaudreault, N., Gilbert, T. L., Griffin, F., Hargrave, P., Howard, R., Huang, L., Jewell, S., Keller, N., Knoblich, U., Larkin, J. D., Larsen, R., Lau, C., Lee, E., Lee, F., Leon, A., Li, L., Long, F., Luviano, J., Mace, K., Nguyen, T., Perkins, J., Robertson, M., Seid, S., Shea-Brown, E., Shi, J., Sjoquist, N., Slaughterbeck, C., Sullivan, D., Valenza, R., White, C., Williford, A., Witten, D. M., Zhuang, J., Zeng, H., Farrell, C., Ng, L., Bernard, A., Phillips, J. W., Reid, R. C., and Koch, C. 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.

Ding, F., Denain, J.-S., and Steinhardt, J. Grounding Representation Similarity with Statistical Testing, 2021. URL http://arxiv.org/abs/2108.01661.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, 2021. URL http://arxiv.org/abs/2010.11929.

Dunefsky, J., Chlenski, P., and Nanda, N. Transcoders Find Interpretable LLM Feature Circuits, November 2024.

Dyballa, L. and Zucker, S. W. 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.

Dyballa, L., Rudzite, A. M., Hoseini, M. S., Thapa, M., Stryker, M. P., Field, G. D., and Zucker, S. W. 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.

Feather, J., Khosla, M., Murty, N. A. R., and Nayebi, A. Brain-Model Evaluations Need the NeuroAI Turing Test, 2025. URL http://arxiv.org/abs/2502.16238.

Flamary, R., Courty, N., Gramfort, A., Alaya, M. Z., Boisbunon, A., Chambon, S., Chapel, L., Corenflos, A., Fatras, K., Fournier, N., Gautheron, L., Gayraud, N. T., Janati,

H., Rakotomamonjy, A., Redko, I., Rolet, A., Schutz, A., Seguy, V., Sutherland, D. J., Tavenard, R., Tong, A., and Vayer, T. POT: Python optimal transport. 22(1):78:3571–78:3578, 2021. ISSN 1532-4435. URL https://dl.acm.org/doi/10.5555/3546258.3546336.

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

Ghosh, A., Chorghay, Z., Bakhtiari, S., and Richards, B. A. 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.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., 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.

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

Höfling, L., Tangemann, M., Piefke, L., Keller, S., Franke, K., and Bethge, M. Only brains align with brains: Cross-region alignment patterns expose limits of normative models. *arXiv preprint arXiv:2604.21780*, 2026.

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

Jacobs, M., Fel, T., Hakim, R., Brondetta, A., Ba, D., and Keller, T. A. Block-Recurrent Dynamics in Vision Transformers, 2026. URL http://arxiv.org/abs/2512.19941.

Khaligh-Razavi, S.-M. and Kriegeskorte, N. 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.

Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., and Krishnan, D. Supervised Contrastive Learning, March 2021.

Kornblith, S., Norouzi, M., Lee, H., and Hinton, G. Similarity of Neural Network Representations Revisited, 2019. URL http://arxiv.org/abs/1905.00414.

Kriegeskorte, N., Mur, M., and Bandettini, P. A. 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.

Lappalainen, J. K., Tschopp, F. D., Prakhya, S., McGill, M., Nern, A., Shinomiya, K., Takemura, S.-y., Gruntman, E., Macke, J. H., and Turaga, S. C. 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.

Lindsey, J., Templeton, A., Marcus, J., Conerly, T., Batson, J., and Olah, C. Sparse crosscoders for cross-layer features and model diffing. *Transformer Circuits Thread*, 2024. URL https://transformer-circuits.pub/2024/crosscoders/index.html.

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

Mémoli, F. 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.

Nellen, N. S., Turishcheva, P., Vystrčilová, M., Sridhar, S., Gollisch, T., Tolias, A. S., and Ecker, A. S. Learning to cluster neuronal function, 2025. URL http://arxiv.org/abs/2506.03293.

Ostrow, M., Eisen, A., Kozachkov, L., and Fiete, I. Beyond Geometry: Comparing the Temporal Structure of Computation in Neural Circuits with Dynamical Similarity Analysis, 2023. URL http://arxiv.org/abs/2306.10168.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., et al. Pytorch: An imperative style, high-performance deep learning library. *CoRR*, abs/1912.01703, 2019. URL http://arxiv.org/abs/1912.01703.

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

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

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

Schrimpf, M., Kubilius, J., Hong, H., Majaj, N. J., Rajalingham, R., Issa, E. B., Kar, K., Bashivan, P., Prescott-Roy, J., Geiger, F., Schmidt, K., Yamins, D. L. K., and DiCarlo, J. J. Brain-Score: Which Artificial Neural Network for Object Recognition is most Brain-Like?, 2020. URL https://www.biorxiv.org/content/10.1101/407007v2.

Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., and Paluri, M. A Closer Look at Spatiotemporal Convolutions for Action Recognition, 2018. URL http://arxiv.org/abs/1711.11248.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., 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.

Wang, E. Y., Fahey, P. G., Ding, Z., Papadopoulos, S., Ponder, K., Weis, M. A., Chang, A., Muhammad, T., Patel, S., Ding, Z., Tran, D., Fu, J., Schneider-Mizell, C. M., da Costa, N. M., Reid, R. C., Collman, F., Franke, K., Ecker, A. S., Reimer, J., Pitkow, X., Sinz, F. H., and Tolias, A. S. 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.

Williams, A. H., Kim, T. H., Wang, F., Vyas, S., Ryu, S. I., Shenoy, K. V., Schnitzer, M., Kolda, T. G., and Ganguli, S. 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.

Williams, A. H., Kunz, E., Kornblith, S., and Linderman, S. W. Generalized Shape Metrics on Neural Representations, 2022. URL http://arxiv.org/abs/2110.14739.

A. Extended Related Work

RSA (Kriegeskorte et al., 2008) and CKA (Kornblith et al., 2019) have become central tools for the comparison of neural representation, distilling high-dimensional population responses into tractable similarity measures that facilitate large-scale cross-system analyses. However, by construction, they operate on population-level summaries of activity, potentially obscuring the internal structure and functional heterogeneity of the underlying units. This is reflected in a growing body of work questioning their statistical and theoretical properties: Ding et al. (2021); Ahlert et al. (2024); Bo et al. (2025); Höfling et al. (2026) show that common measures frequently disagree, while Murphy et al. (2024) identify a bias that drives CKA toward its maximum in the low-sample, high-dimensional regime typical of neural data. On the theoretical side, Harvey et al. (2024) show that CKA, CCA, and Procrustes all quantify alignment of optimal linear readouts, exposing their decoding-centric nature, and Almudévar & Ortega (2026) unify them under a usable-information framework. Avitan & Golan (2025) show in model-recovery experiments that the highest-scoring model by linear readout is frequently not the correct generative model, and Feather et al. (2025) propose requiring models to match internal neural representations within individual variability margins rather than aggregate decoding scores. 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). We emphasize that the decoding-centric — or linear-readout — nature of RSA, CKA, and Procrustes is thus already established in this prior work, and that these metrics were never intended to assert mechanistic or circuit equivalence. Our contribution is therefore not this observation itself, but two constructive additions: (i) a controlled, *causal* dissociation in which encoding topology is manipulated by design while decoding metrics remain matched (Section 3.1), and (ii) an encoding-side measure — the encoding manifold compared via Gromov–Wasserstein distance — that captures the internal organization to which these metrics are blind. Our work is thus 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.

B. Limitations

First, the MNIST experiment uses a specific clustering loss and a relatively shallow architecture; whether the same decoupling holds for deeper networks or other training ob-

jectives remains open. Nevertheless, it provides a concrete existence proof that encoding and decoding metrics can be dissociated, motivating further investigation. The clustering loss also alters tuning sharpness and inter-neuron connectivity simultaneously; an ablation that isolates encoding-manifold topology from the regularizer's direct effect on weight magnitudes (e.g. matching tuning sharpness via L1 or dropout) is left to future work. Furthermore, GW is used here as an intrinsic measure of encoding-manifold coverage and has not yet been validated against ground-truth circuit differences in a cross-system, model–brain comparison, which we leave to future work. Finally, although GW provides a practical measure of encoding similarity, its behavior in more diverse settings remains to be validated in future work.

C. Additional Results

C.1. Encoding Manifold Position Determines Decoding Fidelity

To link subpopulation decoding fidelity to encoding manifold coverage, in Figure 9, 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 D.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 both stimulus classes (see Figures 10 and 11 for 1% and 10% fractions). In this setting where decoding metrics can capture differences between the high-PC1 and low-PC1 conditions, GW can also capture this effect. The high-PC1 subpopulations are significantly closer to the full population as measured by GW self similarity of each subset of neurons to the full set (Figure 9 J). However, this difference is less than in many of the decoding metrics, backing up the claim that these metrics are complimentary.

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). Since there are no known functional groups of neurons in mouse V1 (Dyballa et al., 2024; Nellen et al., 2025), the encoding manifold continuously organizes V1 neurons. These stable biological properties of computation are robustly captured by the encoding manifold pipeline (Appendix D.4, Figure 7). Therefore, we argue that encoding manifold coverage and topology are relevant properties for functional alignment. Our artificial MNIST example provides additional evidence for these claims in a controlled setting (Section 3.1). As a result, sampling region-constrained subpopulations on the encoding manifold omits biologically relevant functions, such as throwing away the majority of retinal ganglion cell types. Still, these functionally misaligned populations yield behaviorally aligned representations, showing the dissociation between functional and behavioral alignment.

C.2. FPS Subpopulation Showcases

Figures 12–26 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 4 reports all eight metrics numerically across all 17 datasets.

C.2.1. Biological Systems

C.2.2. Machine Learning Models

D. Methods

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

D.1. Data

D.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).

D.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

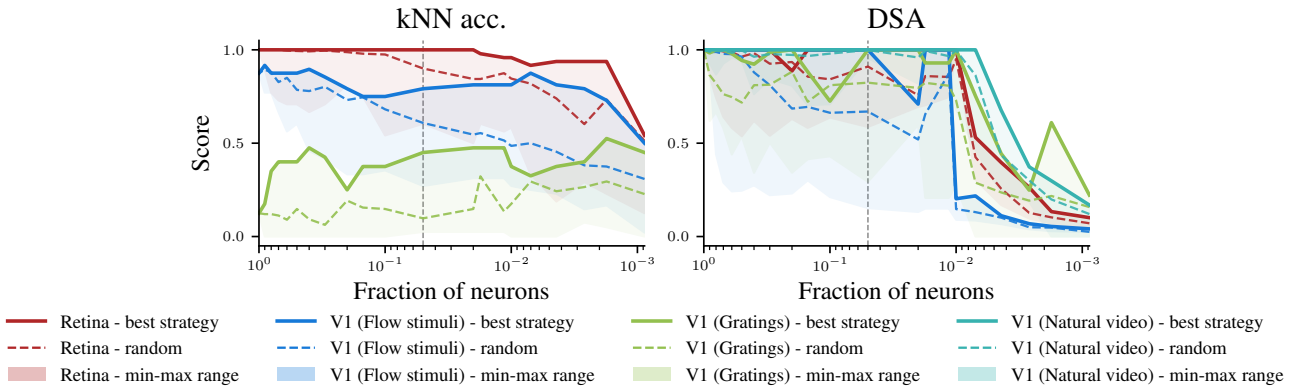

Figure 6. Fraction sweep for kNN accuracy and normalized DSA z scores.

Table 2. 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.

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

full 8-direction $\times$ 5-temporal-frequency drifting grating sweeps ($\sim$ 15 repeats per condition) and natural movie stimuli.

Drifting gratings. For each of the six 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 $> 50\ \mu$V). 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, $\sim$ 900 frames at 30 fps) is segmented into 156 non-overlapping scenes of $\sim$ 6 frames each. Trial-averaged responses are computed per scene in the same 50 ms bins, yielding tensors of shape $(N_{\text{area}} \times 156 \times 1 \times T_{\text{scene}})$.

D.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 27).
- **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

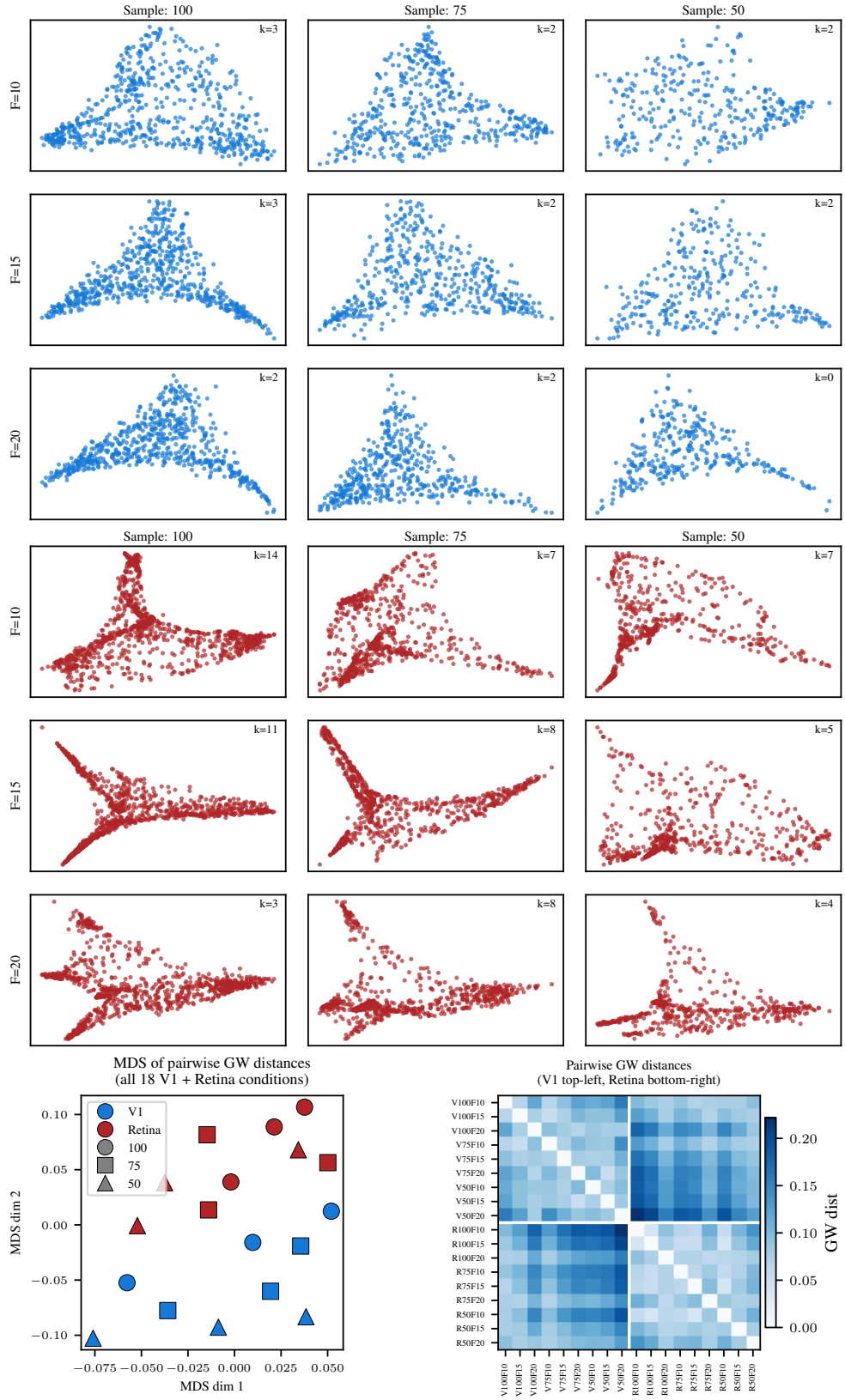

Figure 7. **Encoding manifold pipeline stability** across sample size (columns), tensor decomposition factors (rows) and datasets (Blue: V1, Red: Retina). Encoding topology is stable, showing continuous manifolds in V1 and clustered ones in retina. GW distance groups the two datasets together (bottom).

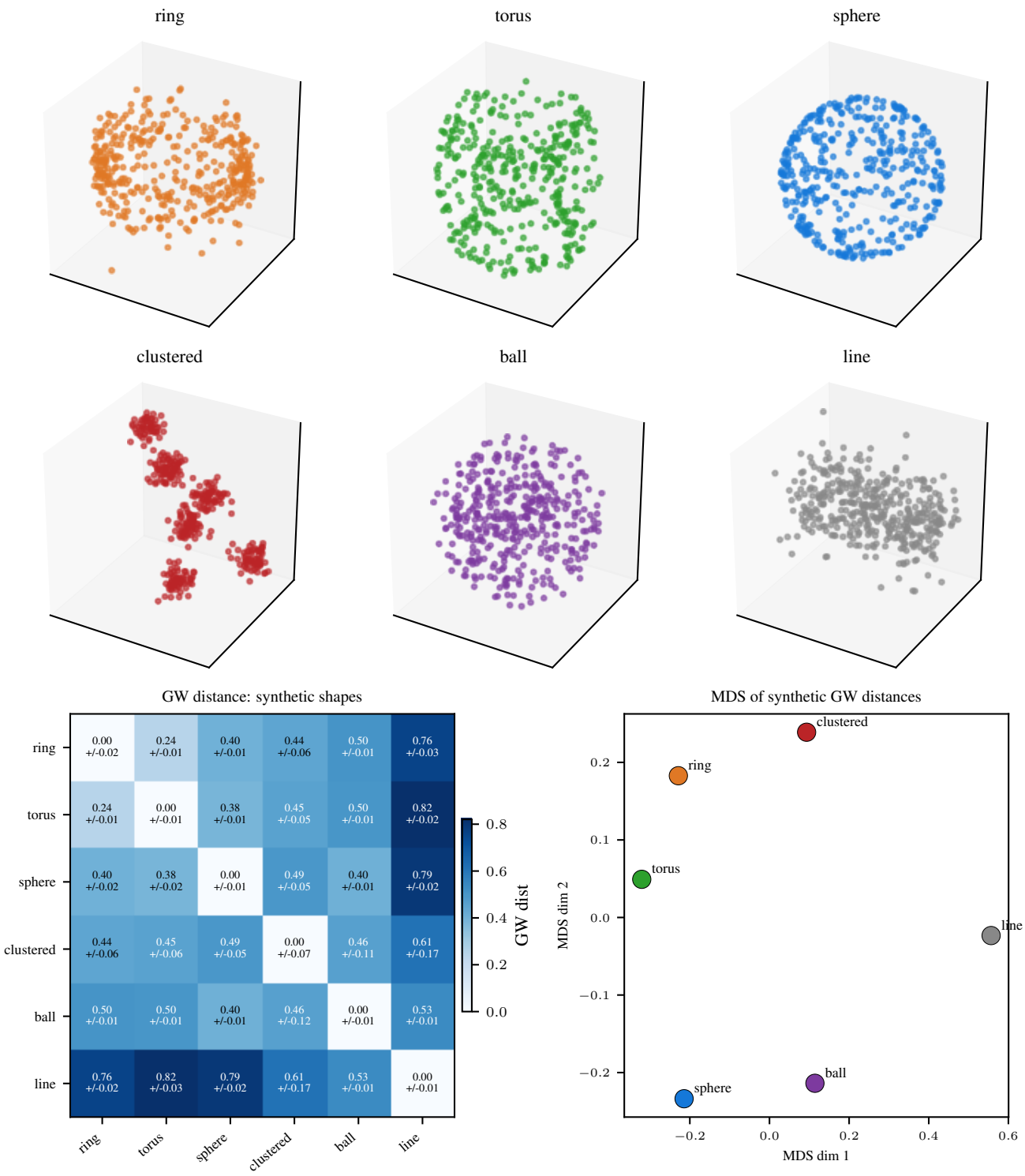

Figure 8. **GW intuition** based on simple synthetic datasets. The ring and torus being topologically equivalent are grouped together. Similarly, the sphere and ball are close in GW space. The line is furthest from all other datasets, not allowing for circular matching of points.

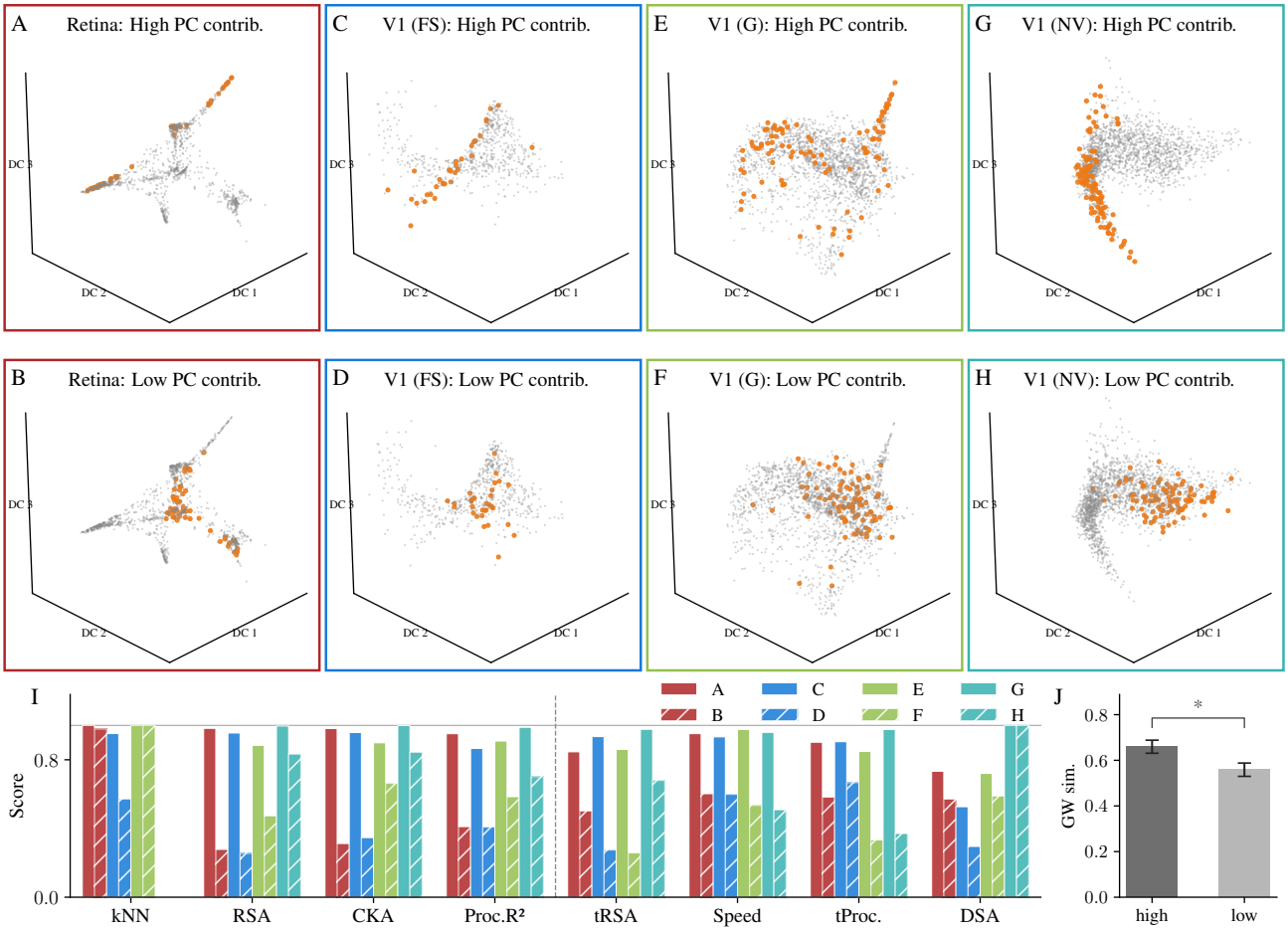

Figure 9. Encoding manifold coverage and decoding scores for 5% subpopulations selected by highest (top row, solid bars) or lowest (bottom row, hatched bars) first-PC contribution in decoding space, for Retina, V1, VISp (gratings), and VISp (natural). Orange dots show selected neurons on the encoding 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 10 (1%) and 11 (10%) (Wilcoxon rank-sum; * $p<0.05$).

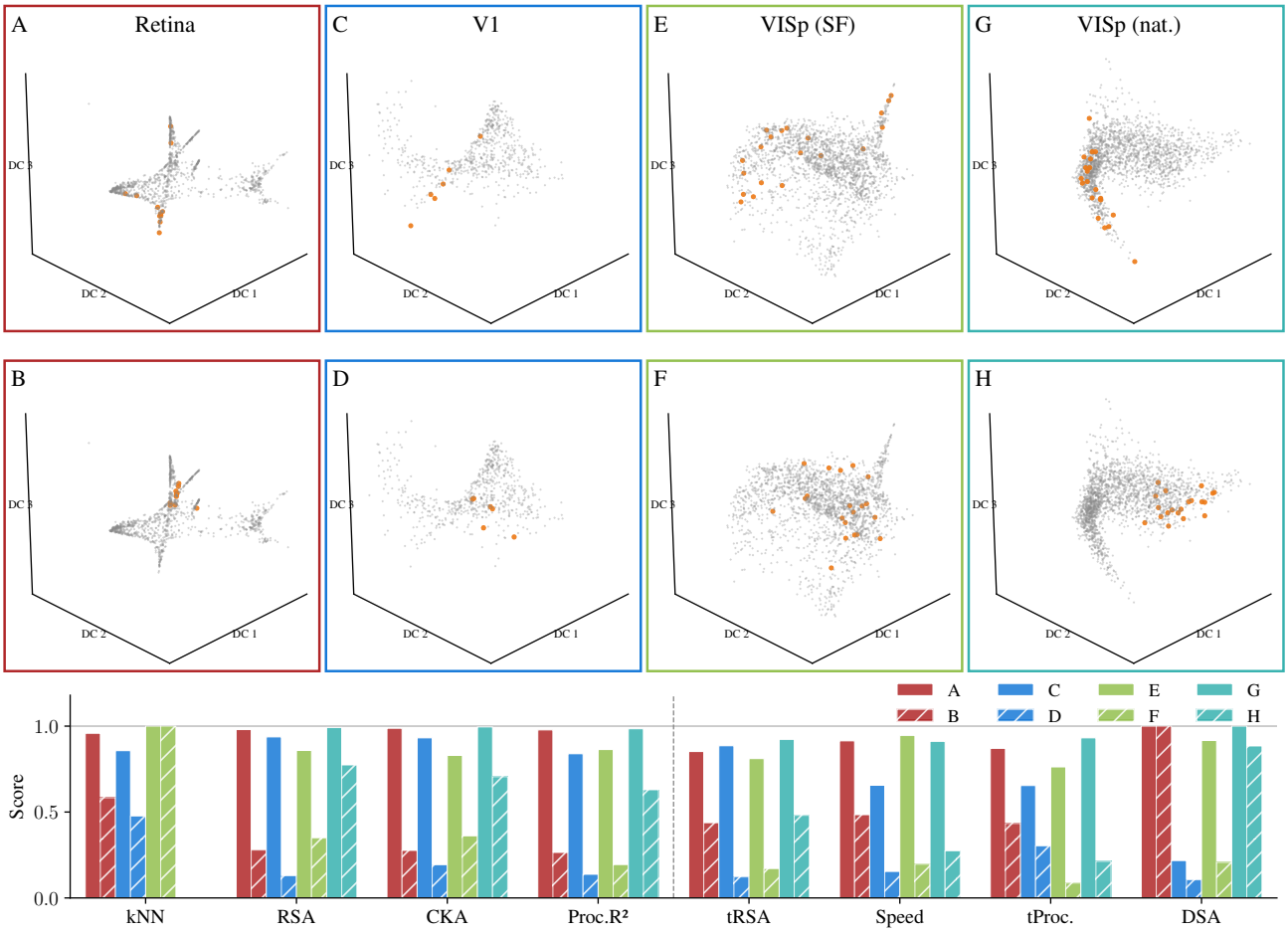

Figure 10. Size-matched encoding manifold region sampling for Retina, V1 and VISp (gratings and natural video). Samples drawn to maximize (top) or minimize (bottom) first principal component contribution of selected neurons, generating near best-/worst-case samples. These samples from distinct regions of the encoding manifold yield qualitatively different decoding profiles under the same eight metrics. In all conditions, a small (1%) subregion sample can suffice to largely recover alignment metrics.

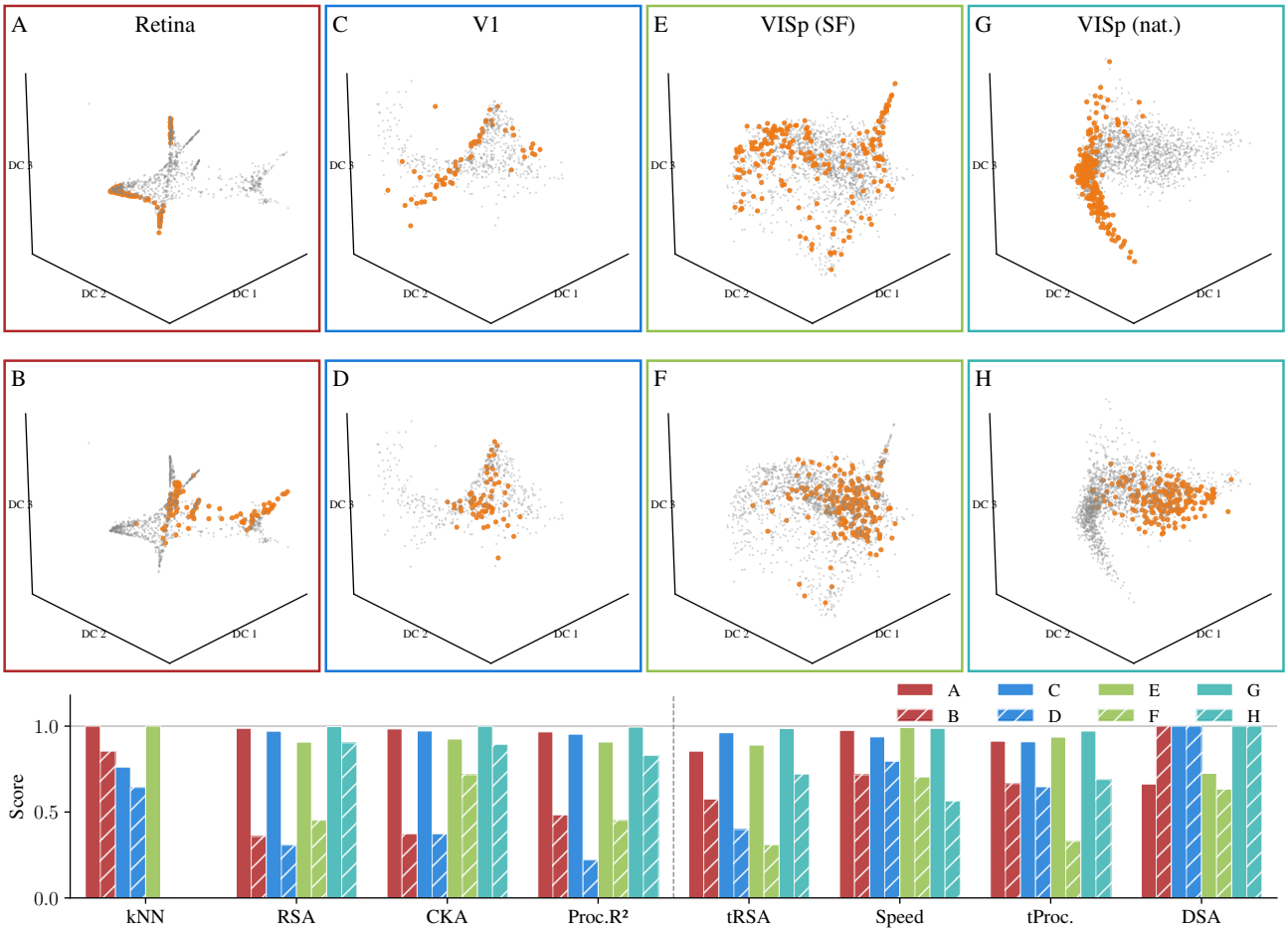

Figure 11. Size-matched encoding manifold region sampling for Retina, V1 and VISp (gratings and natural video). Samples drawn to maximize (top) or minimize (bottom) first principal component contribution of selected neurons, generating near best-/worst-case samples. These samples from distinct regions of the encoding manifold yield qualitatively different decoding profiles under the same eight metrics. In all conditions, a small (10%) subregion sample can suffice to largely recover alignment metrics.

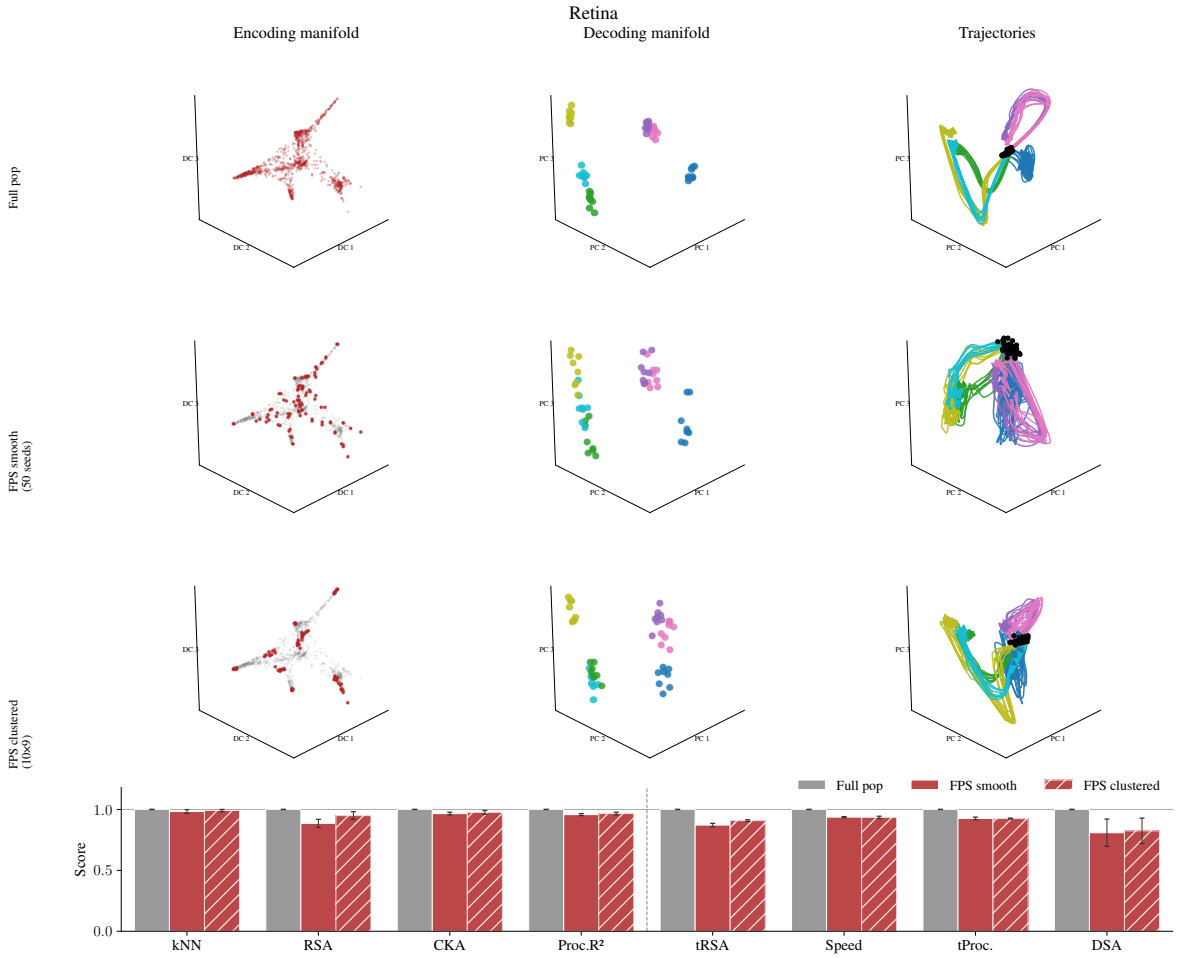

Figure 12. **Retina — FPS subpopulation showcase.** Each row shows the 3-D encoding manifold (left), 3-D decoding manifold (center), and stimulus trajectories (right) for the full population (top), FPS continuous subpopulation (n=50 seeds, m=1 neighbor, ≈100 neurons; middle), and FPS clustered subpopulation (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}).

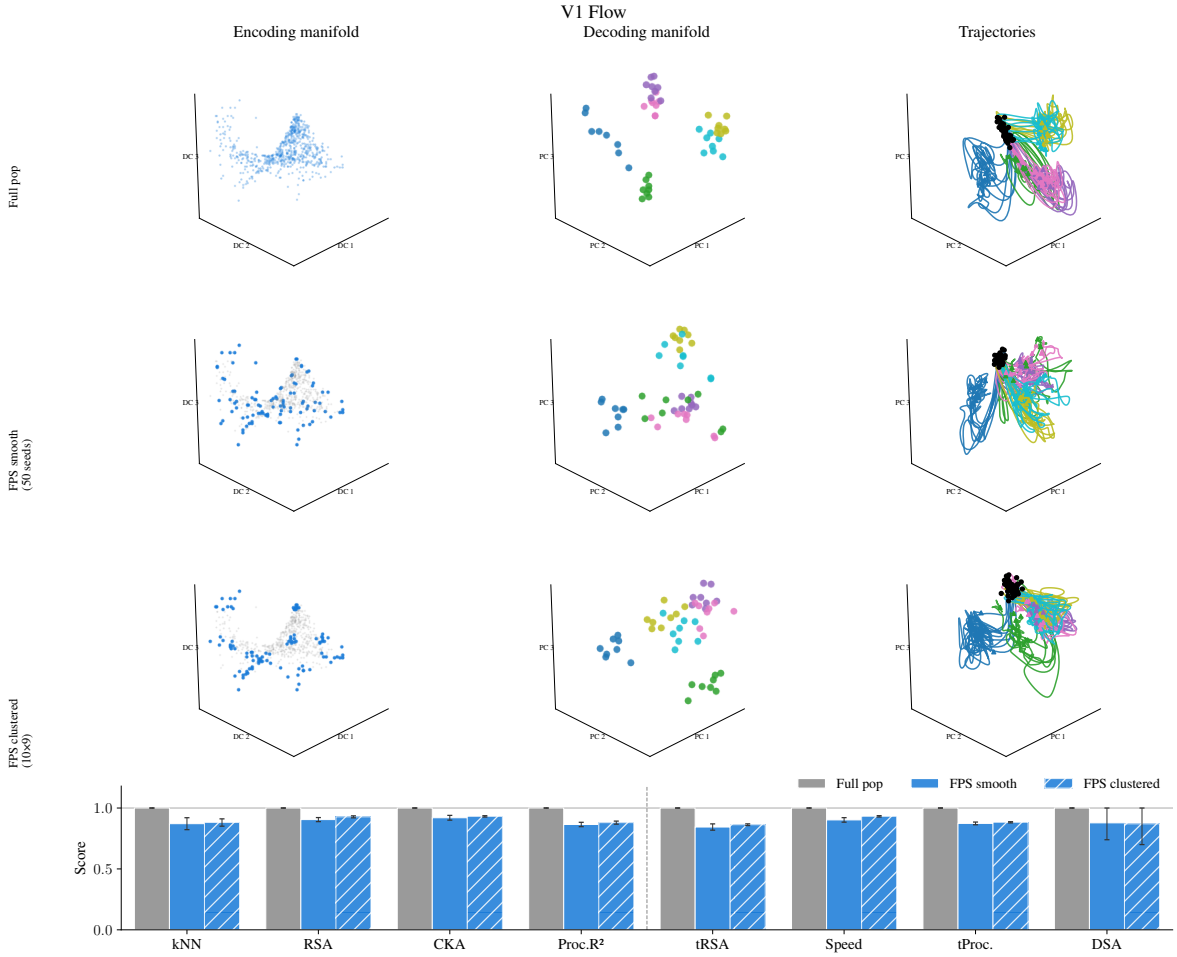

Figure 13. **V1 — FPS subpopulation showcase.** Layout as in Figure 12. FPS continuous: n=50, m=1; FPS clustered: n=10, m=9. Both yield ≈100 neurons.

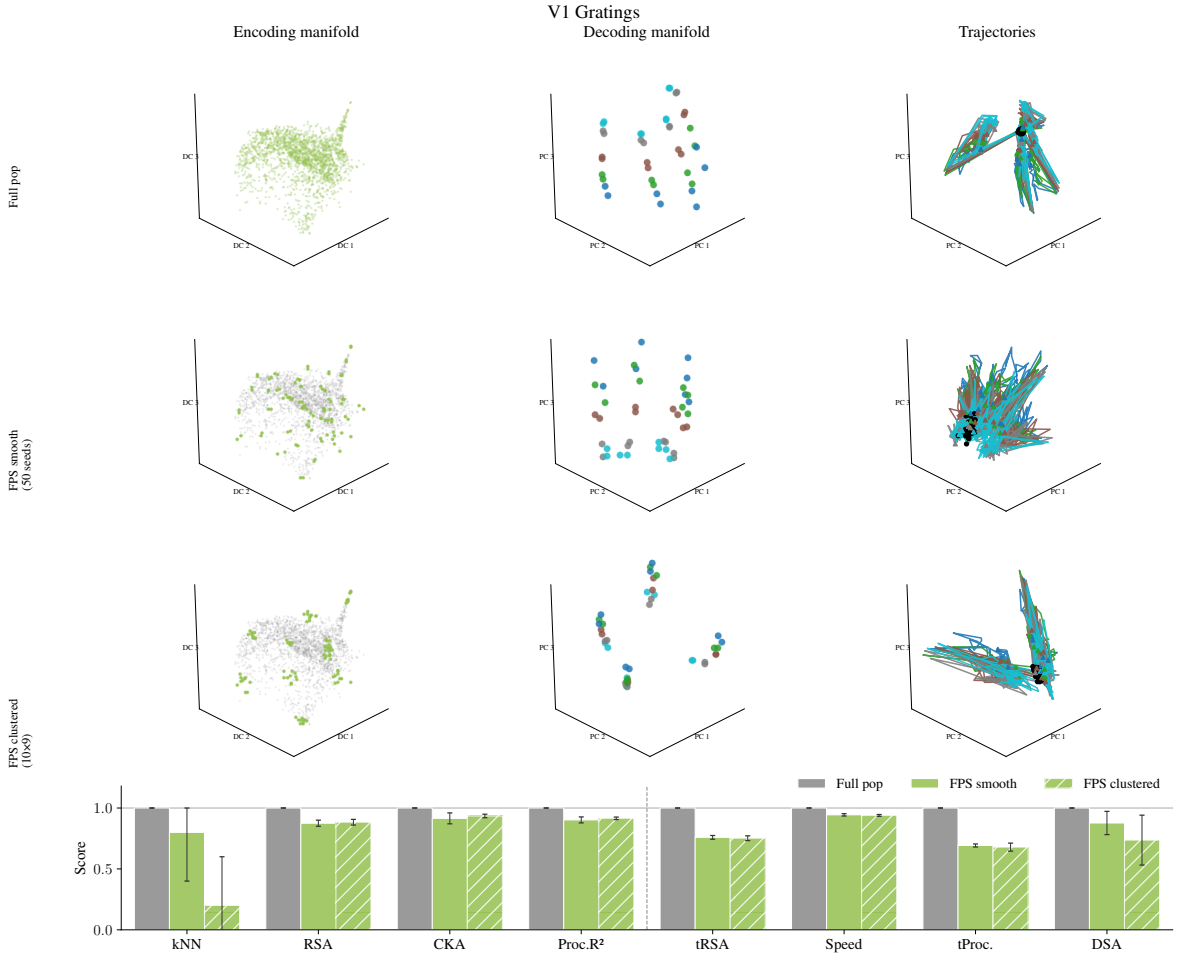

Figure 14. **Allen VISp (drifting gratings) — FPS subpopulation showcase.** Layout as in Figure 12. FPS continuous: n=50, m=1; FPS clustered: n=10, m=9.

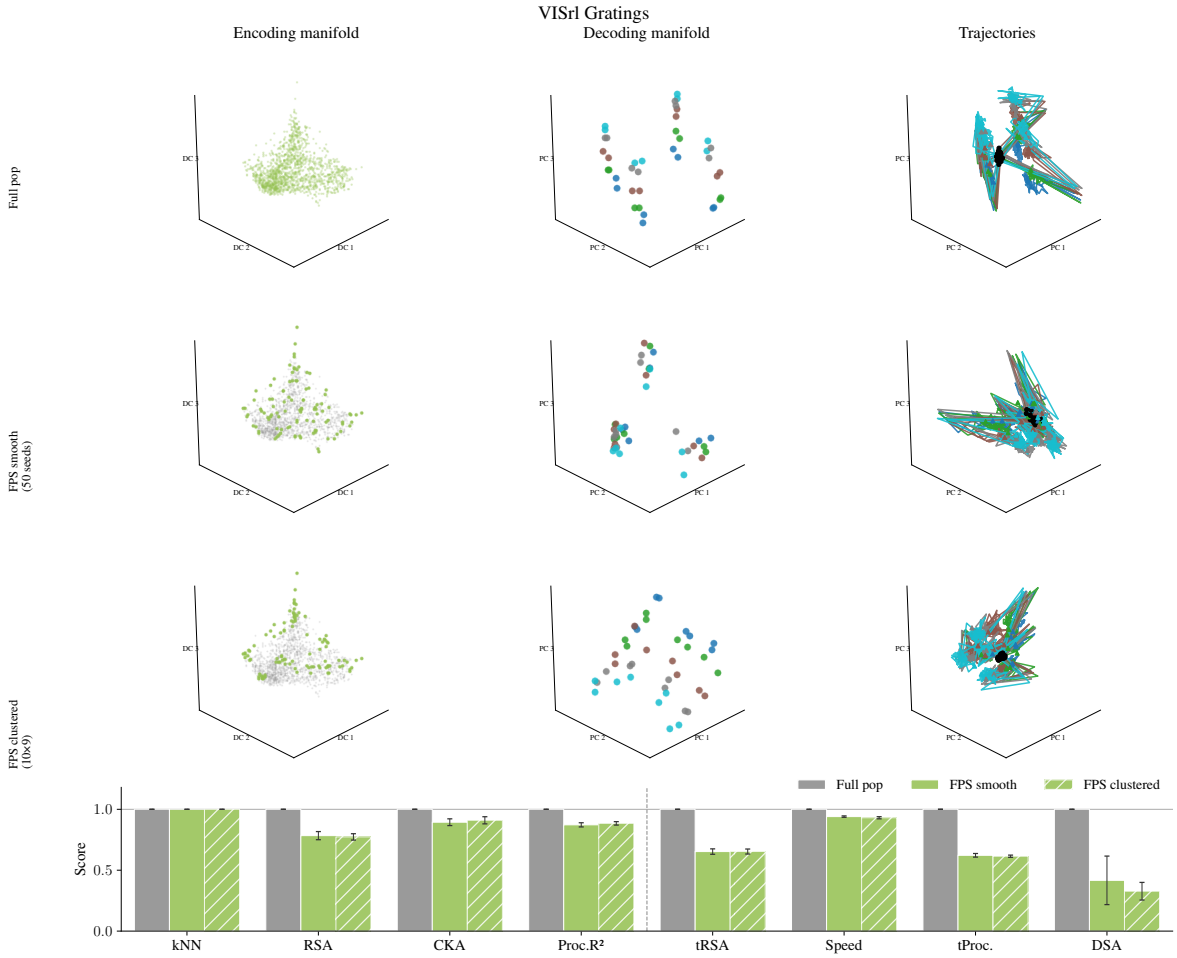

Figure 15. **Allen VISrl (drifting gratings) — FPS subpopulation showcase.** Layout as in Figure 12.

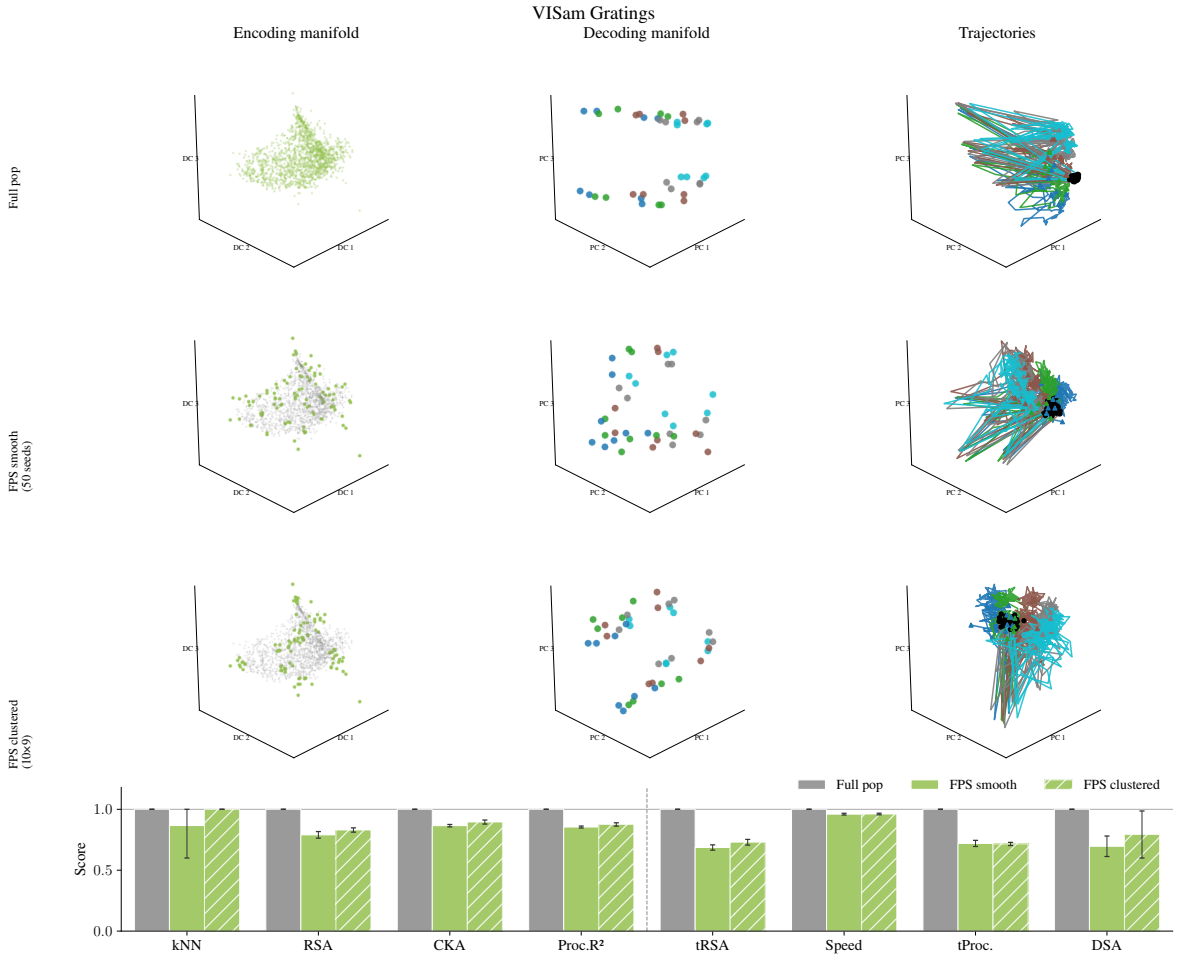

Figure 16. **Allen VISam (drifting gratings) — FPS subpopulation showcase.** Layout as in Figure 12.

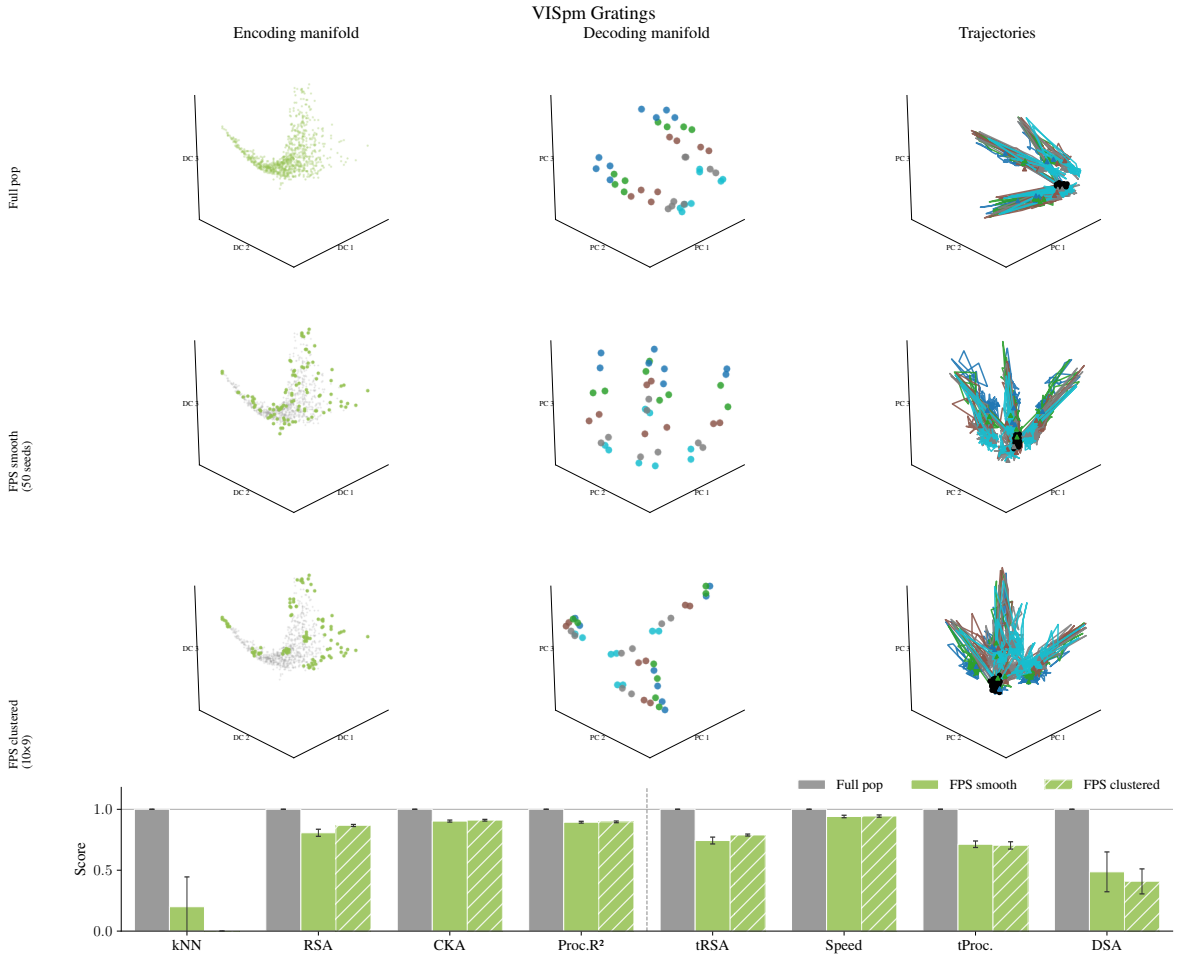

Figure 17. **Allen VISpm (drifting gratings) — FPS subpopulation showcase.** Layout as in Figure 12.

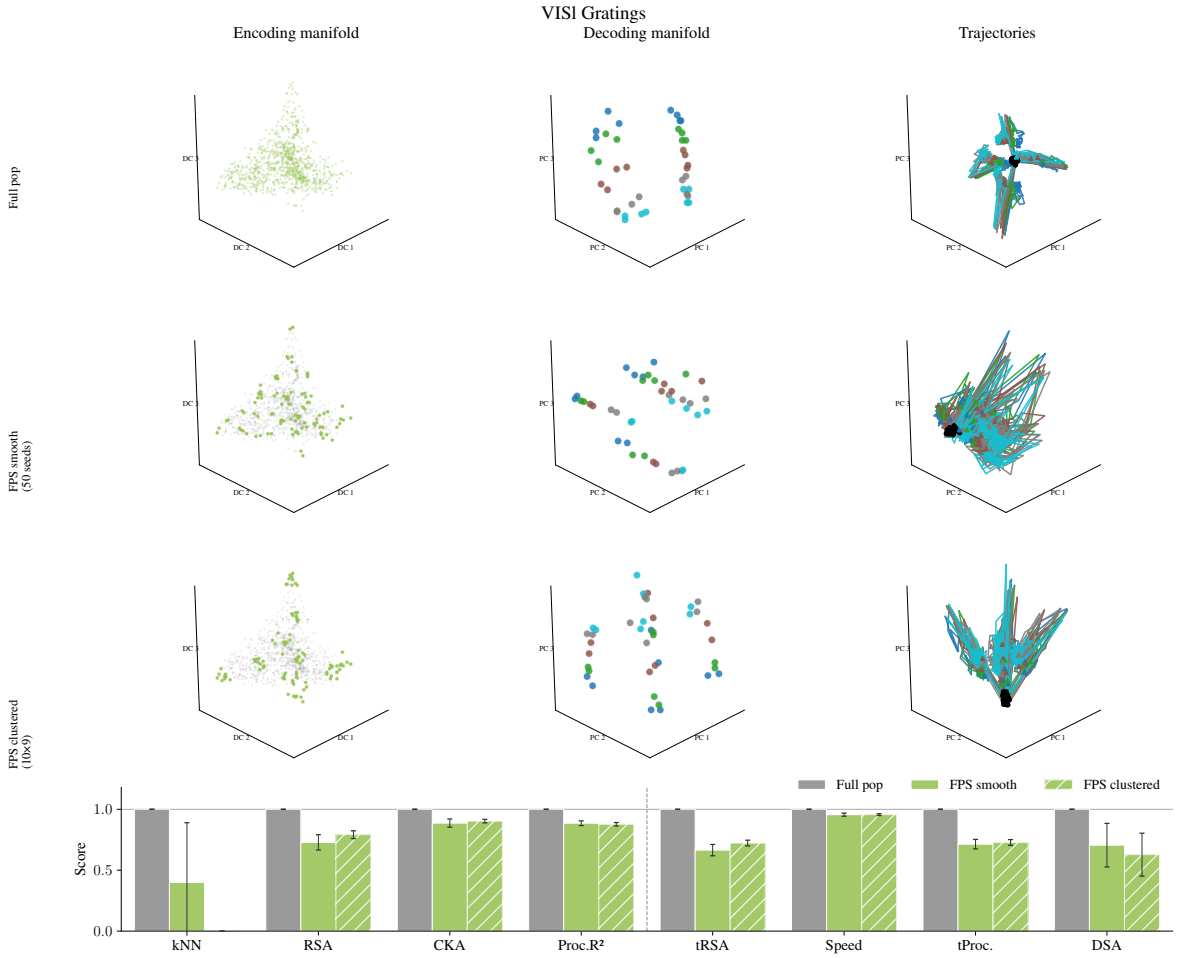

Figure 18. **Allen VISl (drifting gratings) — FPS subpopulation showcase.** Layout as in Figure 12.

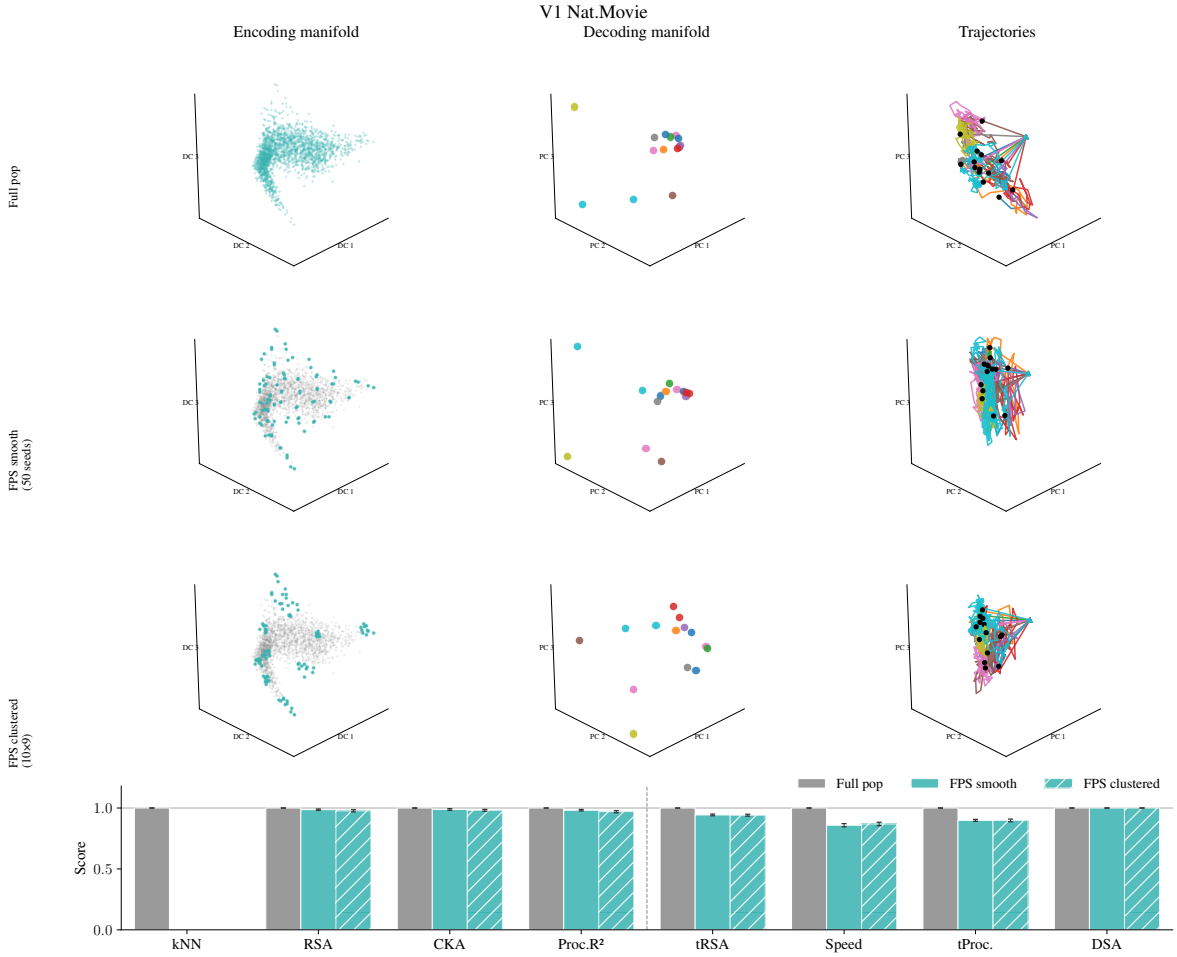

Figure 19. **Allen VISp (natural movie, 156 scenes) — FPS subpopulation showcase.** Layout as in Figure 12. Because the natural movie stimulus has a single direction (D=1), k-NN accuracy is not computed (marked "–" in the bar chart).

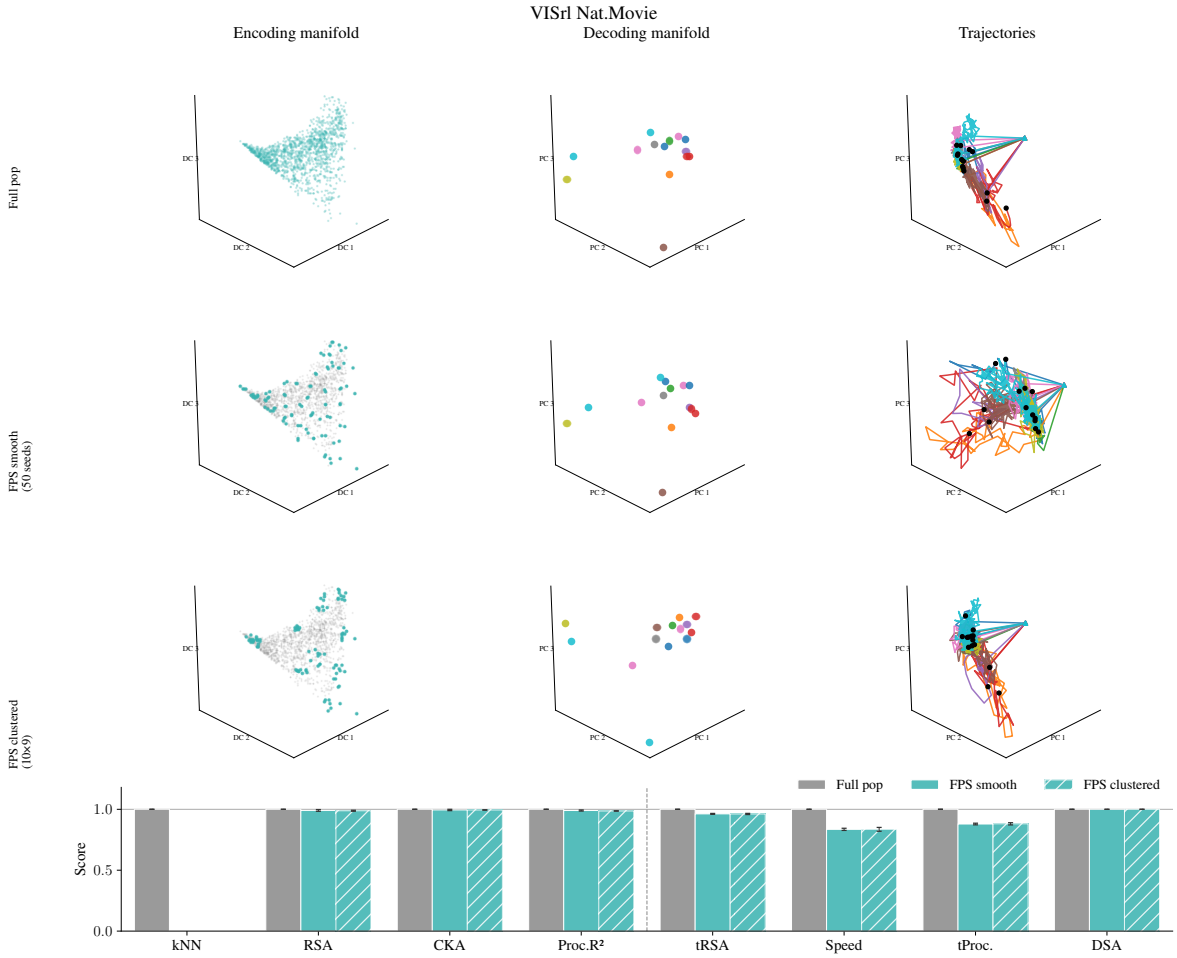

Figure 20. **Allen VISrl (natural movie) — FPS subpopulation showcase.** Layout as in Figure 19.

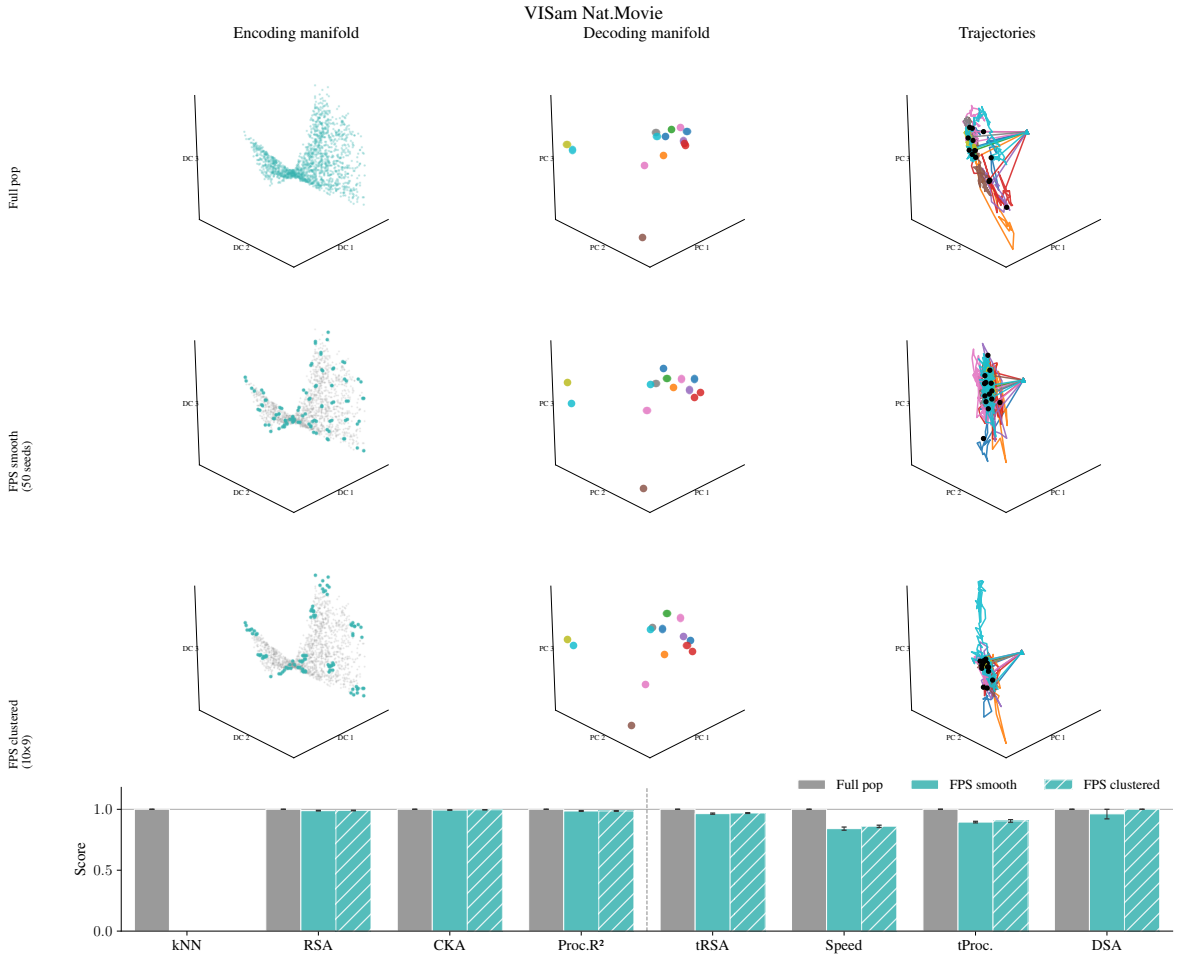

Figure 21. **Allen VISam (natural movie) — FPS subpopulation showcase.** Layout as in Figure 19.

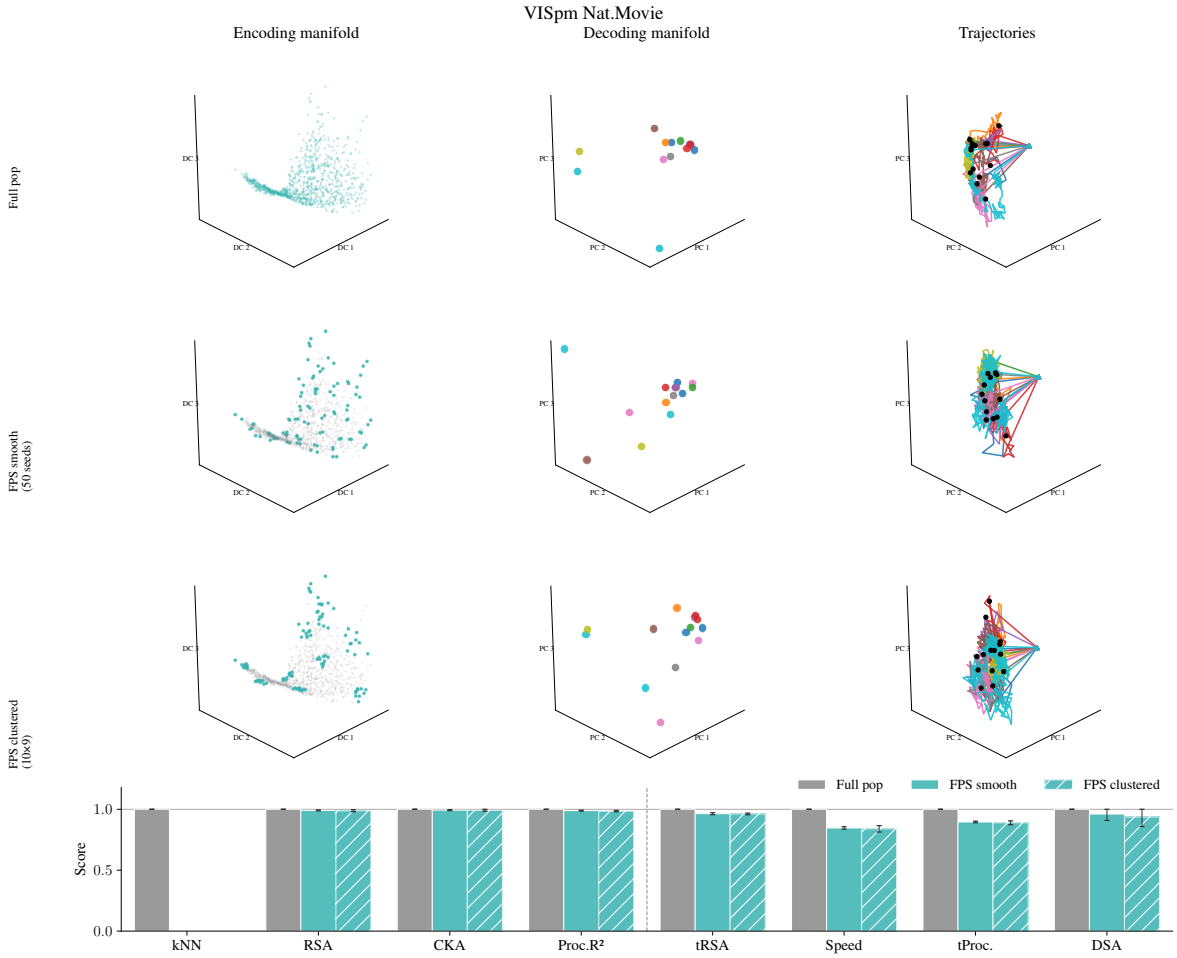

Figure 22. **Allen VISpm (natural movie) — FPS subpopulation showcase.** Layout as in Figure 19.

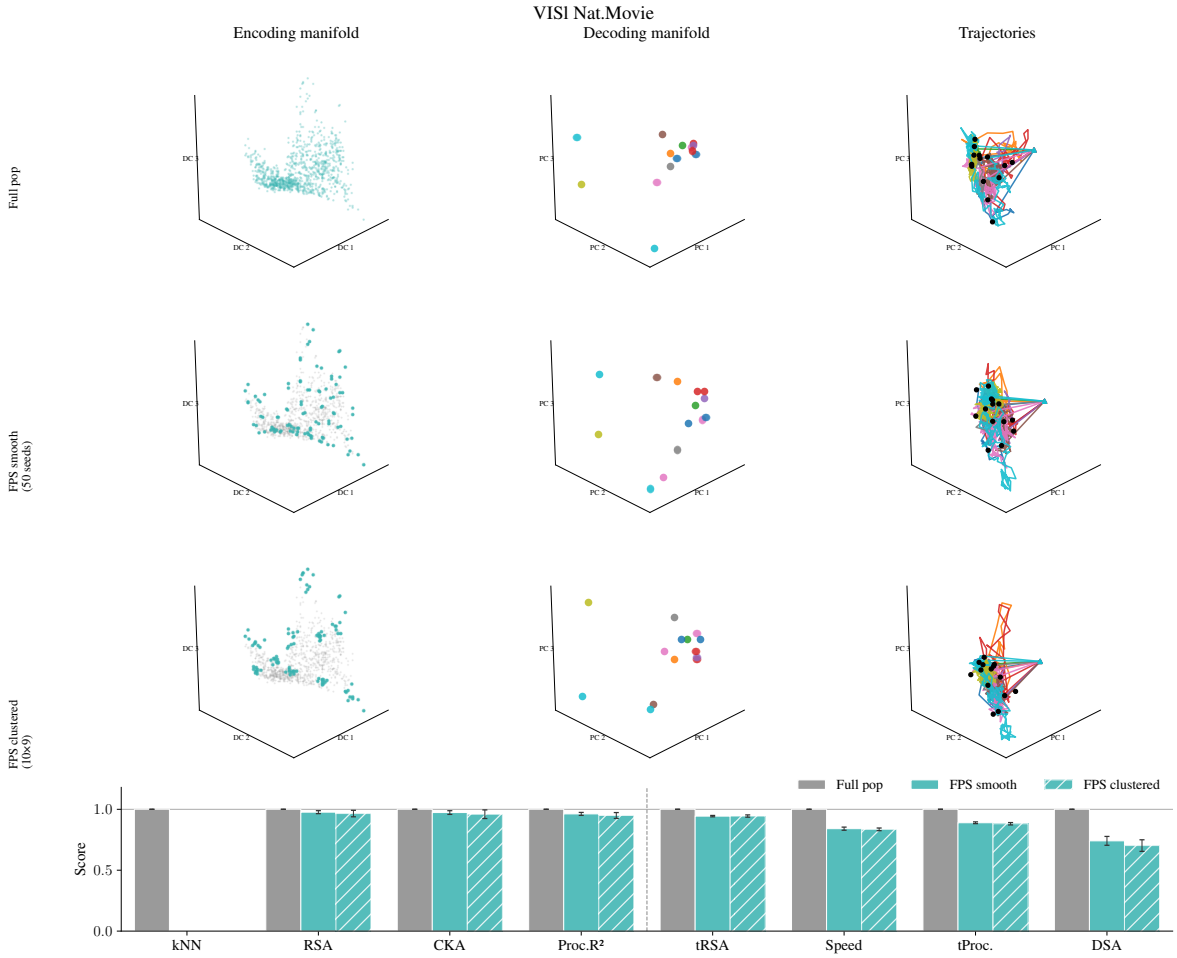

Figure 23. **Allen VISl (natural movie) — FPS subpopulation showcase.** Layout as in Figure 19.

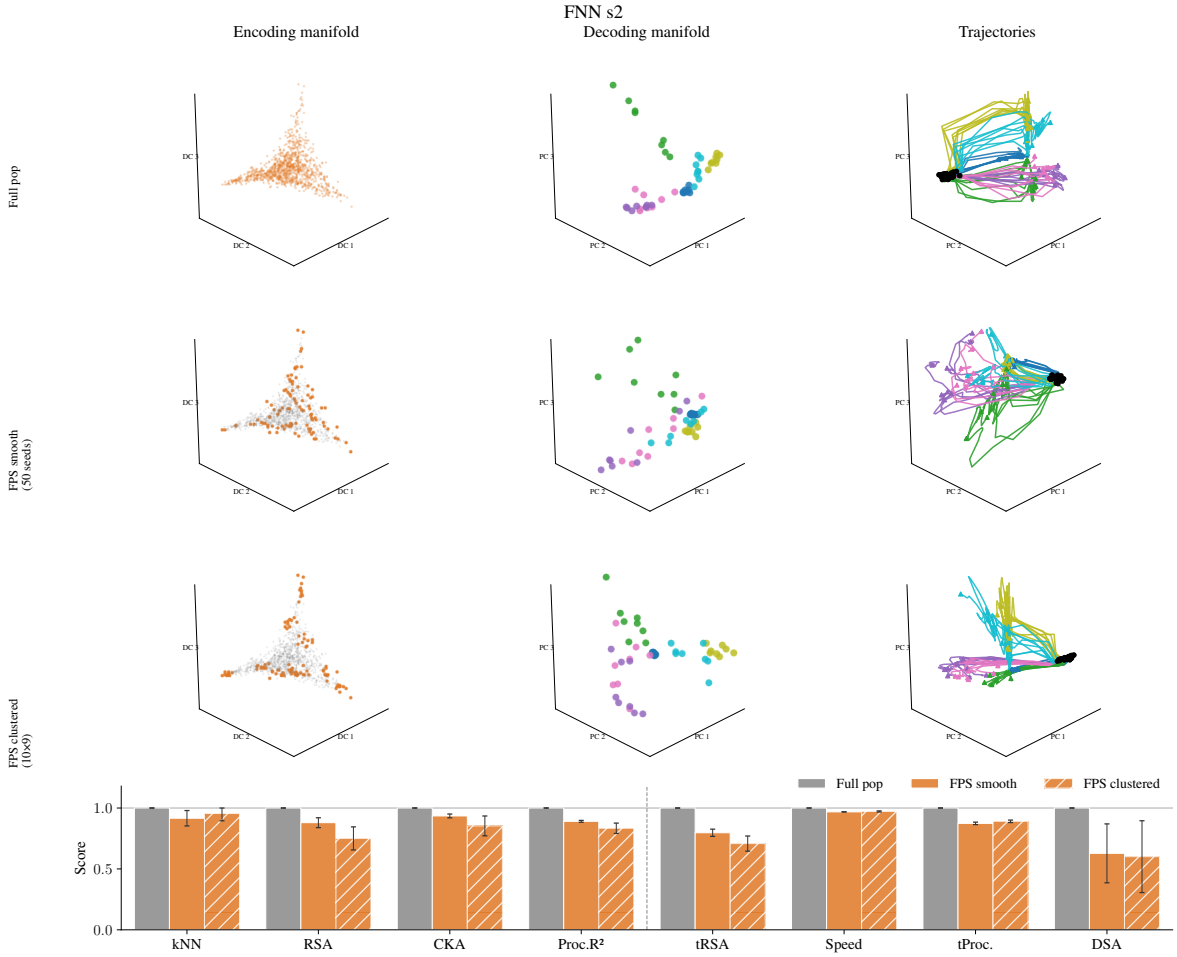

Figure 24. **FNN output layer — FPS subpopulation showcase.** Layout as in Figure 12. 2000 sampled neurons from the FNN recurrent output layer, probed with the 6-class × 8-direction optical-flow stimulus ensemble.

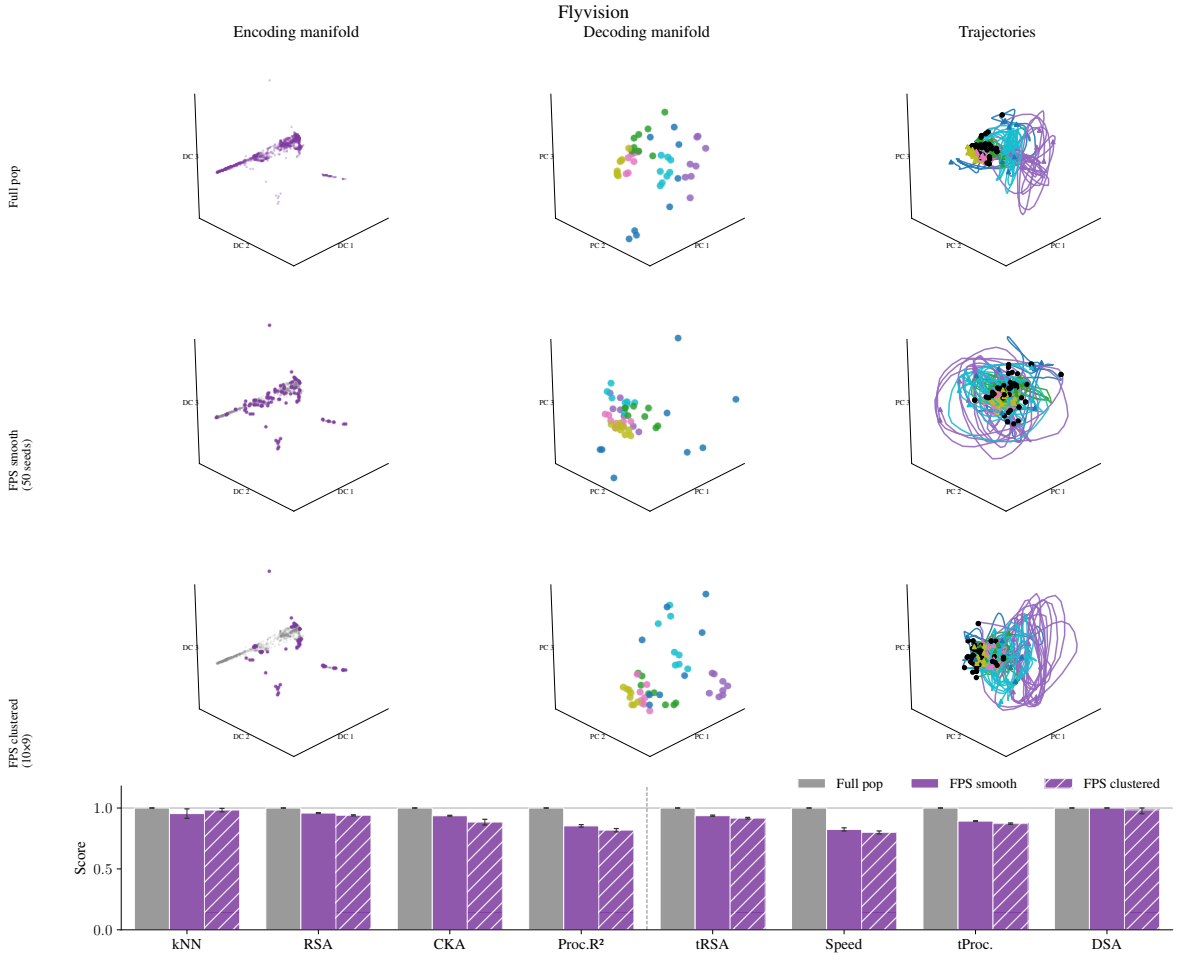

Figure 25. **Flyvision (T, Tm cells) — FPS subpopulation showcase.** Layout as in Figure 12. 1700 T and Tm cell types from a connectome-constrained fly visual system model (Lappalainen et al., 2024).

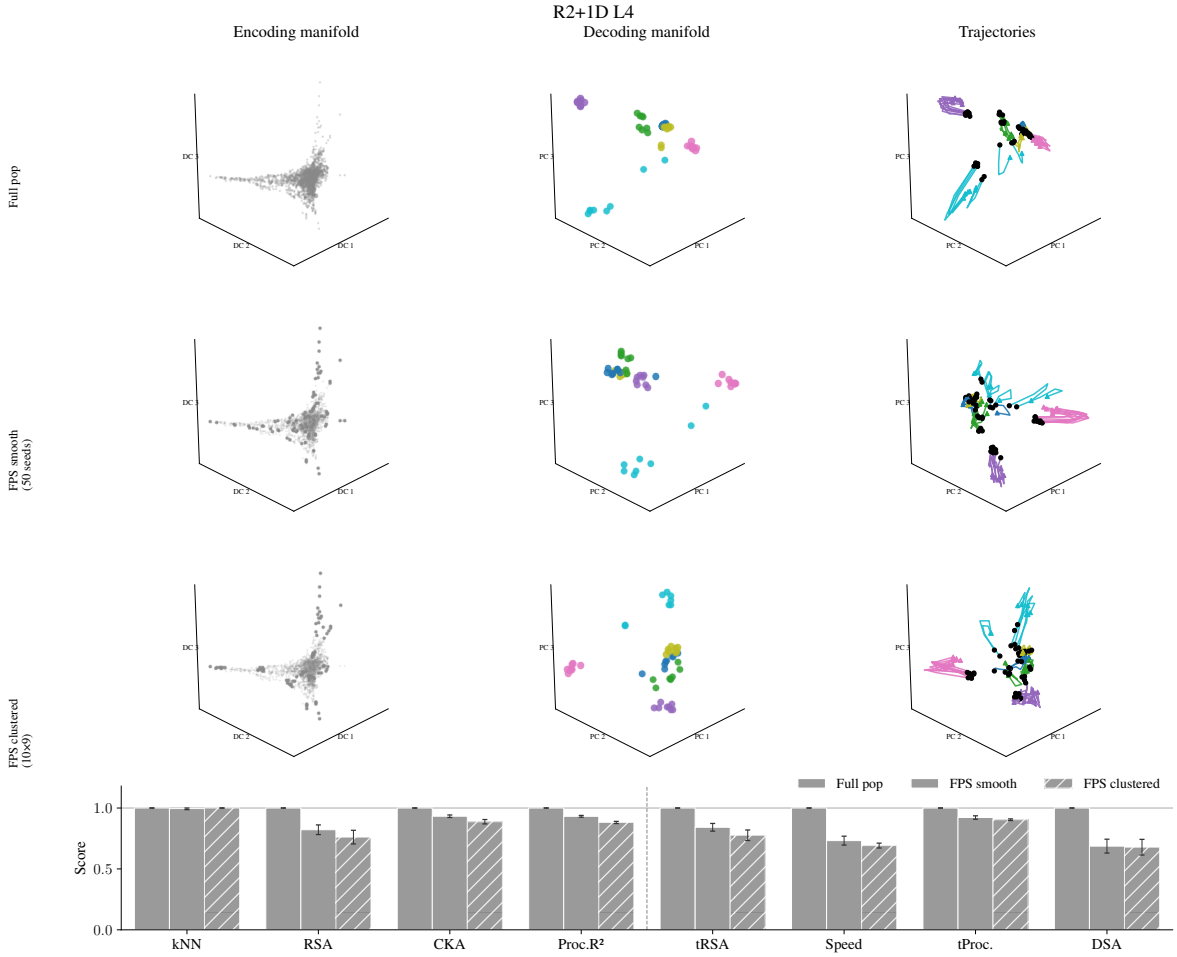

Figure 26. **R(2+1)D layer 4 — FPS subpopulation showcase.** Layout as in Figure 12. 1960 activations from layer 4 of a pretrained spatiotemporal video CNN (Tran et al., 2018).

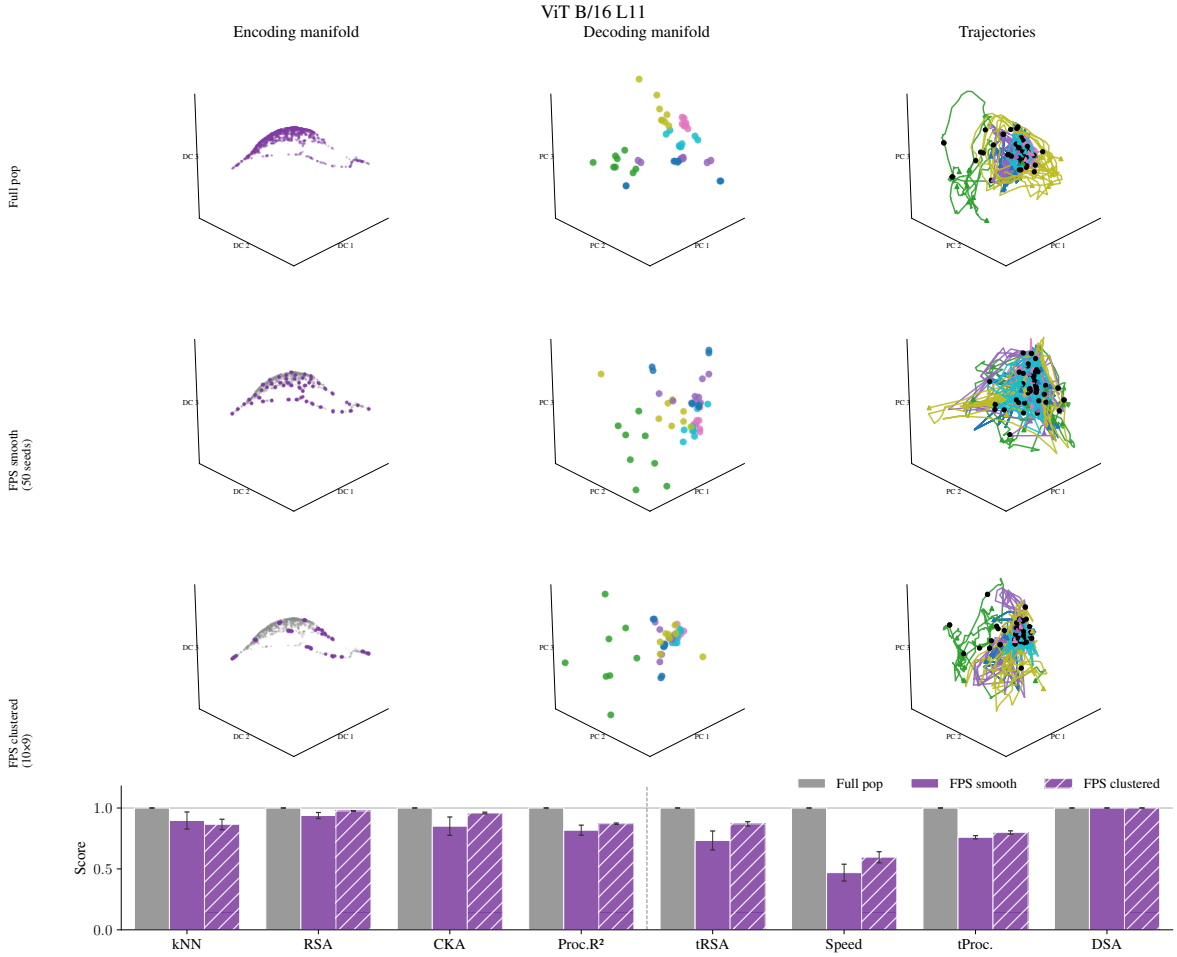

Figure 27. **ViT Layer 11 — FPS subpopulation showcase.** Layout as in Figure 12. Activations from a pretrained vision transformer (Dosovitskiy et al., 2021).

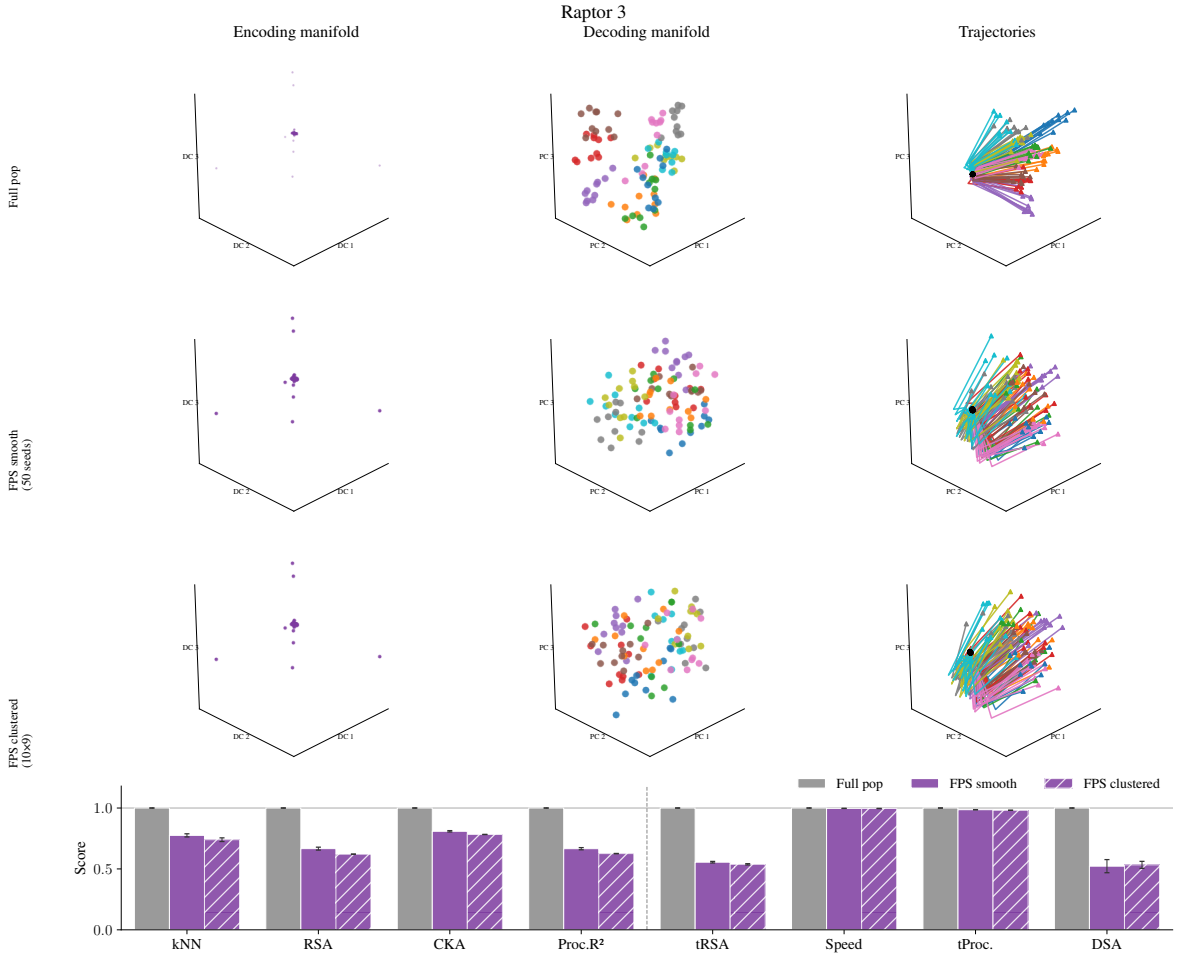

Figure 28. **Raptor — FPS subpopulation showcase.** Layout as in Figure 12. Activations from a pretrained recurrent vision transformer (Jacobs et al., 2026).

Table 3. Gaussian blur robustness: test accuracy at each blur level for baseline (λ=0) vs. clustered (λ=10), n=20 seeds (mean ± SEM, Wilcoxon paired; ns = $p \geq 0.05$).

σ_{blur}	Baseline	Clustered (λ=10)	p
0	0.986 ± 0.000	0.983 ± 0.000	0.004 **
2	0.870 ± 0.011	0.873 ± 0.016	0.846 ns
3	0.621 ± 0.021	0.610 ± 0.026	0.625 ns
4	0.432 ± 0.017	0.387 ± 0.022	0.160 ns
5	0.305 ± 0.017	0.256 ± 0.021	0.064 ns
6	0.216 ± 0.017	0.204 ± 0.020	0.695 ns

are collected across all 12 encoder blocks, as done for the Raptor (Jacobs et al., 2026) model (see Figure 28).

D.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 on 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. rRSA is computed on the full-dimensional space, other metrics on the PCA space.

D.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.

$$\text{Acc}(\hat{X}) = \frac{1}{SK} \sum_{i=1}^{SK} \mathbf{1}[\hat{y}_i = y_i],$$

$$\hat{y}_i = \arg\max_c \sum_{j \in NN_k(i)} \mathbf{1}[y_j = c],$$

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 match. 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 ($m = S * K$):

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

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 $k < 3$ neurons, $P = k$; the smaller manifold is zero-padded to 15 dimensions before alignment). Now solve the orthogonal

Table 4. Subpopulation metric scores across 17 datasets. Top rows (continuous): FPS 50 seeds $\times$ 1 NN; Bottom rows (Clustered): 10 seeds $\times$ 9 NN. Values show mean $\pm$ 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

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.

D.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 per population via `compute_decoding_trajectories`). 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:

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

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

Speed Profile Correlation (SPC).

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

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:

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

Summary scalar: $\overline{R^2_{\text{tProc}}} = \frac{1}{SK}\sum_{s,k} R^2_{\text{Proc},s,k}$. 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 $\mathcal{A}_2$:

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

where d_{DSA} is the DSA distance. We report normalized (against the full population) z-scores of DSA values in comparison to 5 temporally shuffled baseline seeds. Unlike other trajectory metrics, which compare trajectories geometrically, DSA asks whether the two populations implement the *same dynamical transformation* of stimuli over time (Appendix D.9).

D.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.

D.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 ($\sigma_{\text{NTF}} = 3$ 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 (`relNorm` method: divide by each neuron's maximum response, then rescale by relative activation). The resulting 3-tensor has shape $(N \times S \times (K \cdot T))$.

D.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):

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

where $\mathbf{v}^{(1)}_r$ are neural factors, $\mathbf{v}^{(2)}_r$ stimulus factors, and $\mathbf{v}^{(3)}_r$ temporal response factors. Factor vectors are normalized to unit length with magnitudes absorbed by σ^{NTF}_r. 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.

D.3.3. Neural encoding space

Following the reformulation in Dyballa et al. (2024), the scaled neural factor matrix $\mathcal{N}_{\sigma_{\text{NTF}}} = \mathcal{N}\mathbf{\Sigma}_{\text{NTF}}$ (where $\mathcal{N} = \mathbf{X}^{(1)}$ and $\mathbf{\Sigma}_{\text{NTF}} = \text{diag}(\sigma_1^{\text{NTF}}, \ldots, \sigma_R^{\text{NTF}})$) 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 $\mathcal{N}_{\sigma_{\text{NTF}}}$ to reduce dimensionality; the number of retained components is chosen so that cumulative explained variance exceeds 80%.

D.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.

D.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.

D.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:

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

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 $\Psi_t(i) = (\mu_l^t \boldsymbol{\psi}_l(i))_{l=0}^{L-1}$.

D.3.7. MDS visualization

For 3D visualization, metric 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.

D.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 18 conditions per dataset. The pipeline per condition is: tensor4d $\rightarrow$ 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 7).

D.5. Population-size sweep (Section 3.2)

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.

D.6. Encoding manifold region sampling

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 D.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 9J) 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.

D.7. Farthest Point Sampling (Section 3.3)

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 ≈ 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.

D.8. MNIST causal dissociation experiment (Section 3.1)

D.8.1. Architecture

We train a two-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.

D.8.2. 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.

D.8.3. Baseline condition

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

D.8.4. Clustering loss ($\mathcal{L}_{\text{CLUSTER}}$)

For each value of $\lambda \in \{0.001, 0.01, 0.1, 0.5, 1, 5, 10, 50\}$, 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 800-dimensional activation profiles of 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.

D.8.5. Evaluation tensor

For each trained model, a $(800 \times 10 \times 50 \times 1)$ response tensor is constructed: 50 randomly selected test images per digit class, 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.

D.8.6. Encoding manifold construction

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

- CP rank $R = 10$, with `NDIRS` $= 1$ (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).

D.8.7. Encoding similarity (Gromov–Wasserstein)

Encoding manifold similarity is quantified via the Gromov–Wasserstein (GW) distance via entropic regularization (Python Optimal Transport library, `ot.gromov.gromov_wasserstein2`, `square_loss`, max 10 000 iterations, tolerance 10^{-4}).

D.8.8. 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 $\text{attr}_{i,c} = |\partial \ell_c / \partial h_i| \cdot h_i$, 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: $\bar{\cos} = \text{mean}_{c \neq c'}(\hat{\mathbf{a}}_c \cdot \hat{\mathbf{a}}_{c'})$. 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 ($\theta_{\text{FC}} = 0.5$; $\theta_{\text{Conv2}} = 0.3$). Assortativity is computed via NetworkX. Positive values indicate that same-class neurons preferentially connect within class-specific pathways.

D.8.9. Statistical significance

We apply Wilcoxon rank-sum tests with Bonferroni-Holm correction to obtain statistically sound difference between the MNIST trained models.

D.9. 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 Tikhonov regularization $\alpha_{\text{DMD}} = 10^{-2}$.

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.

D.10. 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).

D.11. Reproducibility

All random seeds are fixed and reported: FPS sweep seeds = $\{42, 43, 44, 45, 46\}$, MNIST random initialization seeds = 0–19 for the population-size sweep. Code is available at `https://github.com/JohannesBertram/FNN_Manifolds`, and we provide an interactive Neural Manifold Explorer tool.